

voor vier servo's –
stand-alone of in-system

SUPER- servo- tester

FOCUS ON

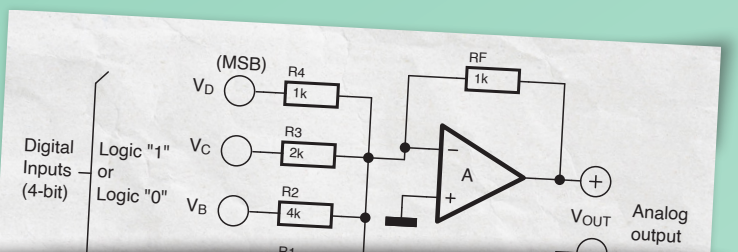
Test & Measurement

Automatisering
van testen en meten
testapparatuur programmeren



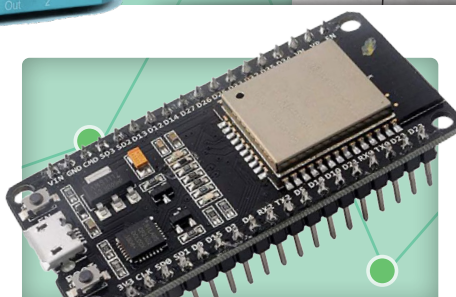
Analoge signalen en
microcontrollers

ADC's, DAC's, stroommeting en meer



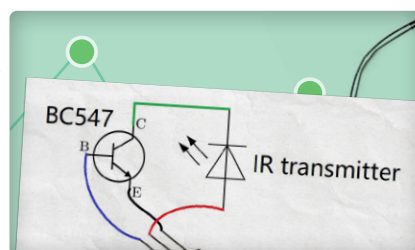
Energielogger
meet en registreert het
stroomverbruik

p. 74



Met Android-smartphone,
spreek ik met ESP32?
praktisch project met de
Android WiFi-API

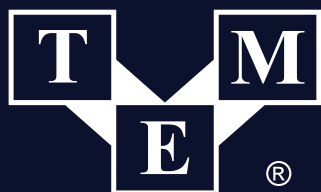
p. 26



IR-afstandsbediening met
smartphone
universeel en aanpasbaar

p. 48





Electronic Components

TRANSFER
MULTISORT
ELEKTRONIK ■ ■

■ GLOBAL DISTRIBUTOR OF ELECTRONIC COMPONENTS ■

Ontdek het assortiment
ElectricVario-schroevendraaiers
en overtuig u ervan dat kwaliteit er toe doet

wiha 
Tools that work for you



Kom meer
te weten:



Transfer Multisort Elektronik B.V.
Croy 7, 5653 LC Eindhoven, tme@tme-benelux.nl

Sluit u bij ons aan:      

tme.eu |

YOU NEED IT, WE HAVE IT!

■ ■ ■ ■ tme.com ■

Elektor verschijnt acht keer per jaar en is een uitgave van

Elektor International Media B.V.

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

www.elektor.nl | www.elektormagazine.nl

Voor al uw vragen
service@elektor.nl

Lid worden
www.elektormagazine.nl/abo

Advertenties
Raoul Morreau
Tel. +31 (0)6 4403 9907
raoul.morreau@elektor.com
www.elektormagazine.nl/adverteren

Auteursrecht
© Elektor International Media B.V. - 2023

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

Druk
Senefelder Misset, Mercuriusstraat 35,
7006 RK Doetinchem (Nederland)

Distributie
Betapress, Nederland – AMP, België

Jens Nickel

Hoofdredacteur Elektor Magazine



Kennis voor allen!

Programmeren is een discipline die ook beginners belooft met het gevoel iets gepresteerd te hebben. Met basiskennis van een programmeertaal en een paar goede ideeën kun je kleine tools maken om taken uit te voeren die anders moeizaam met de hand moesten worden gedaan. En laten we eerlijk zijn: wie heeft zoiets niet trots aan kennis en familie laten zien?

Het programmeren van een kleine database, het opzetten van een server of het implementeren van soortgelijke 'nerdy' dingen worden helaas niet vaak door alle familieleden en vrienden gewaardeerd. De coolness-factor neemt echter toe wanneer iets wordt gemeten, geregeld en bestuurd, vooral wanneer je kunt demonstreren hoe je vondst het dagelijks leven gemakkelijker maakt.

Vooral sommige jonge 'makers' lopen al snel tegen hun grenzen aan. Wie al een paar apps voor de PC heeft geschreven, zal dankzij talen als Python snel de principes van het programmeren van microcontrollers onder de knie krijgen. Maar hoe koppel ik een microcontroller aan sensoren, actuatoren en display-elementen? Waar moet ik aan denken? Overigens zijn ook veel studenten elektrotechniek daar nog niet helemaal zeker van, zoals universitaire docenten ons steeds weer vertellen.

Bij Elektor is het ons doel om zowel beginners als professionals die al tientallen jaren solderen, deze basiskennis bij te brengen. Op pagina 12 vind je een artikel van Mathias Claussen dat beschrijft hoe analoge variabelen (zoals temperatuur) kunnen worden geregistreerd met microcontrollers, en wel met behulp van sensoren die deze variabelen omzetten in spanningen en stromen. Zoals gewoonlijk wordt de praktische kant niet verwaarloosd. Er zijn trouwens ook enkele pareltjes in ons archief over dit onderwerp, bijvoorbeeld de serie "Sensoren voor Arduino en co." van Burkhard Kainka (Elektor november 2016). Zoals altijd kunnen Elektor-leden de PDF's gratis downloaden; en natuurlijk staan de artikelen ook in ons USB-stick archief (www.elektor.nl/20372).

Ook in tijden van gebruiksklare insteekmodules en het automatische genereren van code mag gedegen basiskennis niet verloren gaan. Ik weet zeker dat je het in dit opzicht met me eens bent. Als je verzoeken of suggesties hebt, schrijf me dan op redactie@elektor.com!



Deel met Elektor!

Je elektronica-expertise is welkom! Heb je een voorstel voor een artikel of een elektronica-tutorial op video, of een idee voor een boek? Bekijk onze gids voor auteurs en inzendingen op www.elektormagazine.com/submissions.

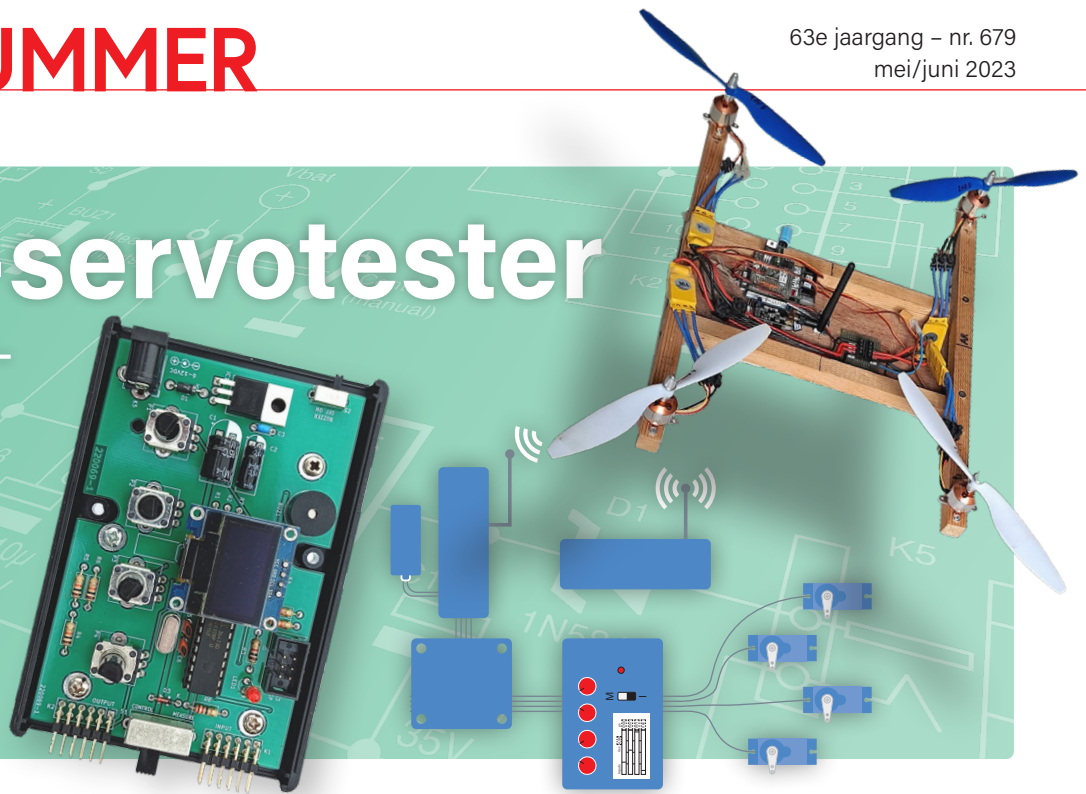
Ons team

Internationaal hoofdredacteur: Jens Nickel | **Redactie:** Asma Adhimi, Eric Bogers, Rolf Gerstendorf, Ton Giesberts, Hedwig Hennekens, Qiëlle Lont, Alina Neacsu, dr. Thomas Scherer, Clemens Valens, Brian Tristram Williams | **Vaste medewerkers:** Roberto Armani, David Ashton, Tam Hanna, Priscilla Haring-Kuipers, Ilse Joostens, prof. dr. Martin Ossmann, Alfred Rosenkränzer | **Vormgeving & Layout:** Harmen Heida, Sylvia Sopamena, Patrick Wielders | **Directeur:** Erik Jansen | **Technische vragen:** redactie@elektor.com

Super-servotester

voor vier servo's –
stand-alone of
in-system

6



Rubrieken

3 Kennis voor allen!

FOCUS

24 Developer's Zone

Sub-Nyquist-sampling in de praktijk

39 Alle begin...

...multivibreert verder!

42 Project 2.0

correcties, updates en brieven van lezers

72 Uit het leven gegrepen

elektronica op hoog niveau

94 Oost West Lab Best

werk in uitvoering...

110 Ethiek in actie

generatieve KI

114 Hexadoku

puzzelen voor elektronici

44 Het nieuwe I3C-protocol

een waardig opvolger van I²C – of gebakken lucht?

52 Microcontroller-documentatie verklaard

deel 2: registers en blokschema's

79 Montage van de M.A.R.S. Rover Kit van 4tronix

92 Achter de schermen van DHZ high-end audio

Ton Giesberts van Elektor over de fijne kneepjes van het analoge ontwerpen

106 Niet voor watjes – Makerfabs Robot Arm Kit

met de Raspberry Pi Pico en MicroPython

Industry

FOCUS

56 Automatisering van testen en meten

testapparatuur programmeren om je te dienen

FOCUS

60 Elektor Infographics

testen en meten

62 Overspanningsbeveiliging voor veilige werking

transiëntbescherming voor niet-geïsoleerde DC/DC-voedings-modules

FOCUS

66 Meetapparatuur van het merk Wiha

testers en meters voor installaties waar u op kunt bouwen

FOCUS

68 Automatisch testen en samenwerken aan testresultaten

Achtergrond en info

FOCUS

12 Analoge signalen en microcontrollers

ADC's, DAC's, stroommeting en meer

20 embedded world 2023

interessante techniek van de beurs

26 Met Android-smartphone, spreek ik met ESP32?

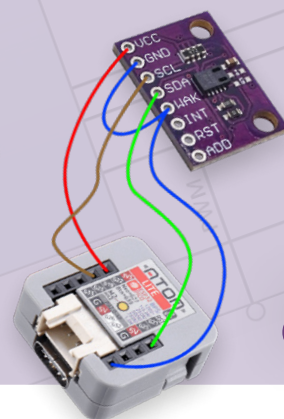
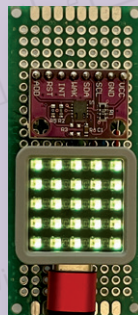
praktisch project met behulp van de Android WiFi-API

Automatisering van testen en meten testapparatuur programmeren



56

eCO₂ Telegram-bot meet de luchtkwaliteit via Telegram



89

Projecten

FOCUS

6 Super-servotester

voor vier servo's – stand-alone of in-system

FOCUS

34 Actief 1kHz-filter voor vervormingsmeting

meet beter door het meetsignaal te optimaliseren

48 BlueRC: IR-afstandsbediening met smartphone en ESP32

universeel en aanpasbaar

FOCUS

74 Energielogger

meet en registreert het stroomverbruik

82 Parkeerschijf met e-papier display

innovatief digitaal alternatief

FOCUS

89 eCO₂ Telegram-bot

meet de luchtkwaliteit via Telegram

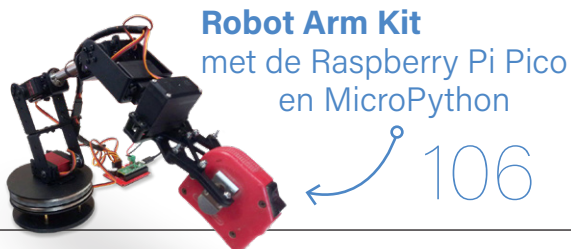
98 RFID-tag's lezen, en een RFID-deurslot

voorbeeldprojecten uit de Elektor Arduino Experimenteerbundel

FOCUS

103 HF-stroomprobe voor oscilloscoop

HF-stroommetingen gemakkelijk gemaakt



Robot Arm Kit
met de Raspberry Pi Pico
en MicroPython

106

Binnenkort

Elektor juli/augustus 2023

Ons volgende reguliere nummer concentreert zich op het Internet of Things en sensoren, en u kunt de bekende mix verwachten van projecten, schakelingen, achtergrondinformatie en alles wat een hobbyist, maker of elektronica-engineer zich maar kan wensen.

Een greep uit de inhoud:

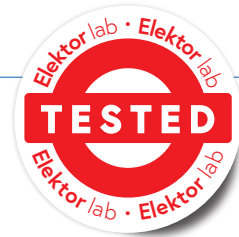
- > Energiezuinig LoRa-weerstation
- > ESP32-gebaseerde impedantie-analyzer
- > Snelheidsmonitor op basis van GPS
- > Doppler-sensor in de praktijk
- > Kiesschijf-telefoon als afstandsbediening
- > Alles over het Matter-protocol
- > Raspberry Pi Pico W NTP-klok met CircuitPython
- > TinyML: een stap-voor-stap handleiding

...en nog veel meer!

Aankondigingen onder voorbehoud.

Elektor juli/augustus 2023 verschijnt omstreeks 12 juli 2023.





Super-servo-tester

voor vier servo's – stand-alone
of in-system



Figuur 1. De quadcopter waarvoor de super-servotester is ontworpen.

Olivier Croiset (Frankrijk)

Ik heb de hier gepresenteerde super-servotester ontwikkeld omdat de drone waar ik aan werk tijdens testvluchten soms een onaangename geur van gebraden elektronica en verschroeide koperlak verspreidt. Ik had een tester nodig om het probleem te lokaliseren en me te helpen begrijpen wat er aan de hand was.

Op internet zijn veel servotesters te vinden waarmee een enkele servo getest kan worden. Daarmee kun je controleren of de servo die je net hebt gekocht goed werkt. De hier voorgestelde tester gaat een paar stapjes verder. Je kunt er niet alleen tot vier servo's tegelijk mee testen, maar ook de Electronic Speed Control-eenheid (ESC) of flight controller samen met servo's in diverse configuraties.

De drone waarvoor ik de super-servotester nodig had, is afgebeeld in **figuur 1**. Hij is gebaseerd op [1].

Servosignalen voor beginners

Ter herinnering: een servo wordt aangestuurd door een PWM-sigitaal met een frequentie (of pulsherhalingsfrequentie, PRR) van 50 Hz. De breedte van de puls (de

tijd dat het signaal hoog is) moet tussen één en twee milliseconden liggen, de rest van de tijd is het signaal laag. De servo staat in zijn middenpositie als de puls 1,5 ms duurt. Als de puls korter is, draait de servo in de ene richting (bijvoorbeeld linksom); als hij langer is, draait hij in de andere richting (bijvoorbeeld rechtsom). In een typisch (vliegend) systeem moeten vijf parameters worden gecontroleerd: de duur van de vier servo-commandopulsen (aandrijving, roll, pitch en yaw) en de voedingsspanning. In het geval van een complex ontwerp, bijvoorbeeld de ontwikkeling van een flight controller (van een drone of een ander RC-model) of wat voor een pulsmixer dan ook kan met deze tester het resultaat van de mixer worden bekeken.

Merk op dat in het vervolg van dit artikel de term 'servo' kan worden vervangen door ESC of elk ander apparaat dat servosignalen produceert en/of herkent.

Functies en mogelijkheden

De super-servotester (SST) meet de duur van de stuurpulsen van maximaal vier servosignalen en geeft informatie over de kwaliteit van de voeding. Hij kan tussen de ontvanger van de afstandsbediening en de servo's worden geplaatst, tussen de ontvanger en de flight controller van de drone, of tussen de flight controller en de servo's. De **figuren 2 tot en met 6** tonen de mogelijke configuraties.

Bedieningsmodi

De SST heeft twee bedrijfsmodi die met een (schuif) schakelaar gekozen kunnen worden:

- **manual:** in deze modus genereert de SST de pulsen voor vier servo's of voor de flight controller. De breedte van de pulsen wordt bepaald met vier potentiometers. In deze modus is het de SST die de servo's (of de ESC) van stroom voorziet en mag de voeding van het model op geen enkele servo aangesloten zijn. De voedingsspanning van de SST moet tussen 7,5 VDC en 12 VDC liggen.
- **inputs:** in deze modus wordt de lengte gemeten van de pulsen die van de ontvanger of controller komen. De signalen worden ook doorgestuurd naar de uitgangen van de SST om de servo's (of de flight controller) aan te sturen. In deze modus worden de SST en de servo's (of ESC's) gevoed door de voeding van het model. Die mag niet hoger zijn dan 7,49 V en de SST mag niet op zijn eigen voeding worden aangesloten. Ook moeten alle vier de ingangskanalen aangesloten zijn, anders geven de LED en de zoemer van de SST een foutmelding.

Displaymodi

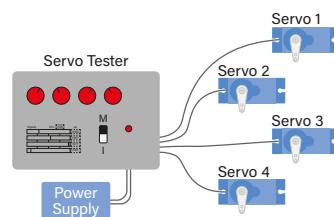
Het display geeft de duur van de pulsen grafisch weer in vier staafdiagrammen, samen met de numerieke waarde in microseconden (van 1000 μ s tot 2000 μ s). De staafdiagrammen beperken zich tot het bereik van 1000...2000 μ s, maar de numerieke waarden niet. Wanneer een waarde buiten de grenzen valt, wordt om die waarde een kader getekend. In dat geval gaat ook de LED branden en zal de zoemer piepen als die is ingeschakeld.

Het display toont ook de waarde van de voedingsspanning van de servo's in beide bedrijfsmodi (**manual** of **inputs**). De kwaliteit van de voeding van het model is namelijk belangrijk voor de flight controller en voor de veiligheid van de piloot en de mensen die zijn/haar vliegkunsten bewonderen. De gemeten voedingsspanning wordt omkaderd wanneer die lager is dan 4,5 V. De LED zal ook oplichten en de zoemer zal piepen, zodat je niet de hele tijd naar de tester hoeft te kijken.

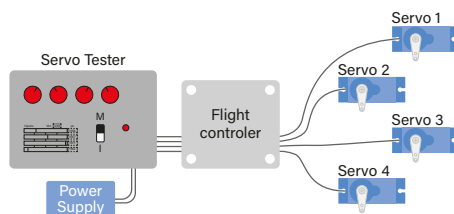
De SST heeft twee weergavemodi, **stack** en **square** (zie **figuur 7**). De tweede modus is beter geschikt voor een quadcopter. De stand van potentiometer P4 bij het inschakelen bepaalt de displaymodus. Als P4 helemaal naar links is gedraaid voor het inschakelen, gebruikt de SST het **stack**-display. Door potentiometer P4 voor het inschakelen naar rechts te draaien, wordt het **square**-display geselecteerd wanneer de tester wordt ingeschakeld.

De schakeling

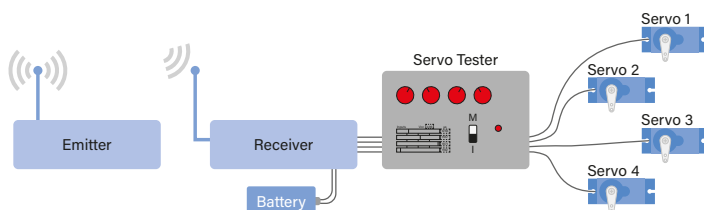
Nu we weten hoe we de tester moeten gebruiken, werpen we eens een blik op de schakeling. Deze is getekend in



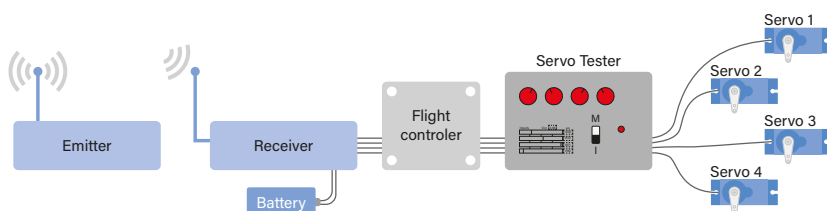
Figuur 2. Configuratie 1 – eenvoudige test van vier servo's (manual modus), maakt het mogelijk vier servo's tegelijk te testen. De servo's worden gevoed door de tester en aangestuurd via de potentiometers.



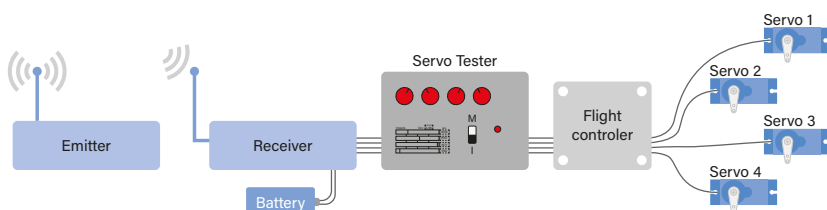
Figuur 3. Configuratie 2 – flight controller-test (manual modus), maakt het mogelijk de flight controller (van een drone) te testen zonder zender of ontvanger. De servo's en de flight controller worden gevoed door de tester en aangestuurd via de potentiometers.



Figuur 4. Configuratie 3 – test van de zender en de ontvanger (inputs modus) om de zender en de ontvanger te controleren. De tester, servo's en flight controller worden gevoed door de batterij van de ontvanger.

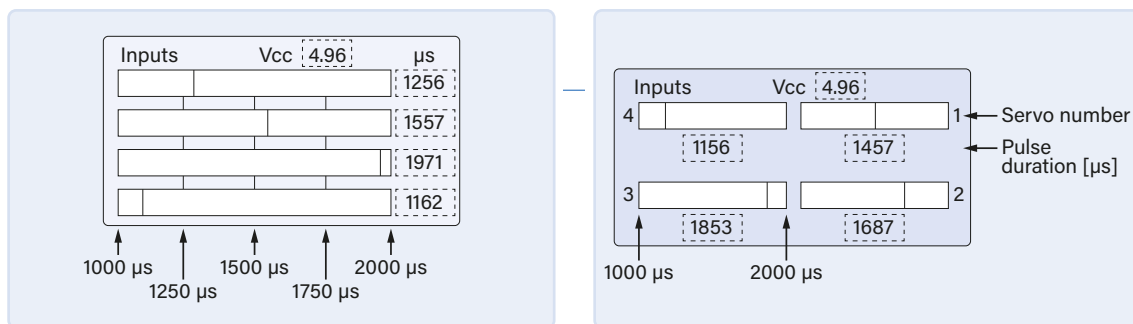


Figuur 5. Configuratie 4 – zender, ontvanger en flight controller test (inputs modus), dient om de goede werking van de zender en de ontvanger met de flight controller te controleren (in het geval van een drone). De tester, servo's en flight controller worden gevoed door de batterij van de ontvanger.



Figuur 6. Configuratie 5 – test van zender en ontvanger (inputs modus), maakt het mogelijk de flight controller te testen met de zender en de ontvanger. De tester, servo's en flight controller worden gevoed door de batterij van de ontvanger.

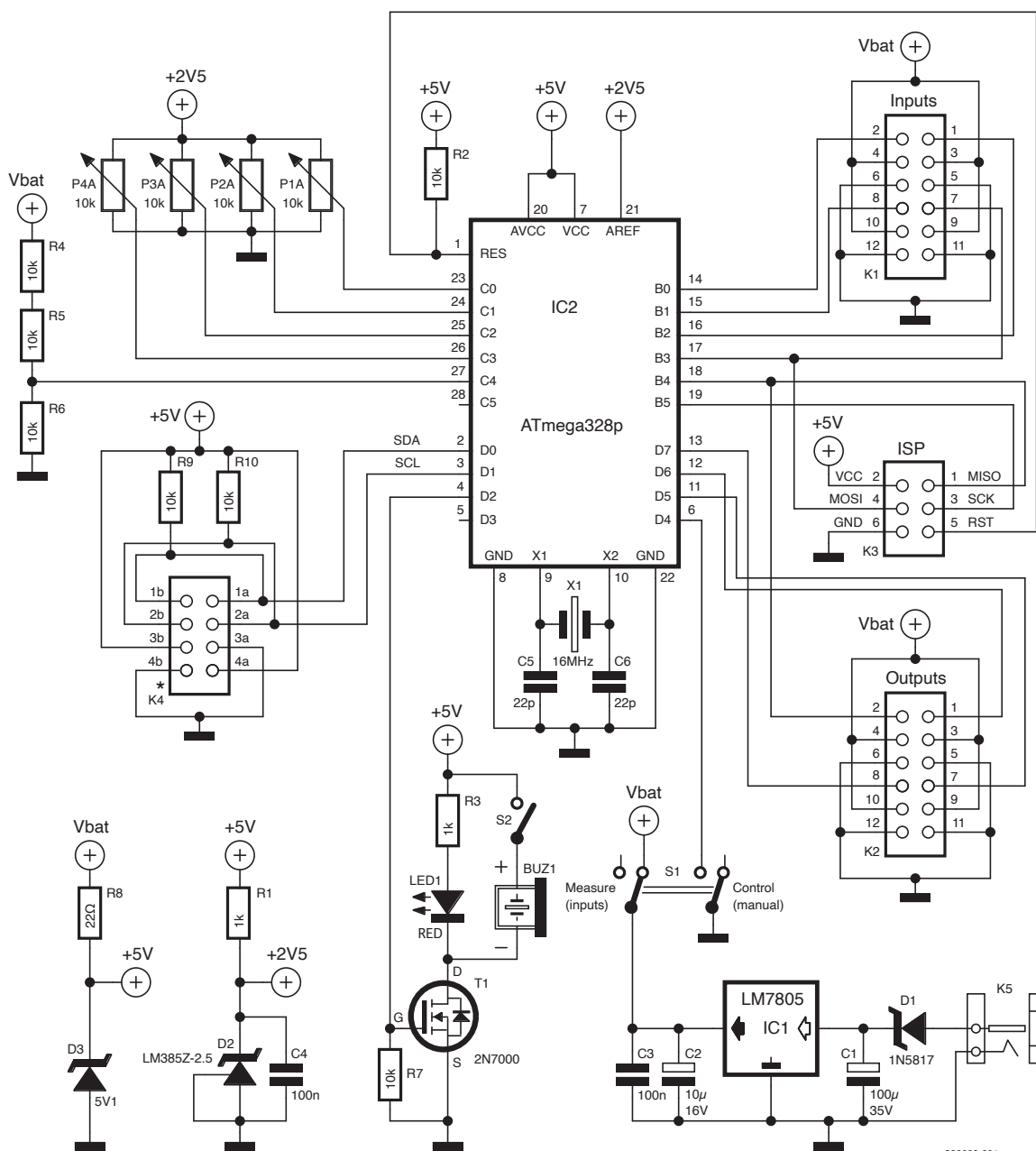
Figuur 7. Het display toont de met schakelaar S1 geselecteerde bedrijfsmodus (inputs of manual), de spanning V_{BATT} (als V_{CC}) en een staafdiagram met numerieke waarden voor elk van de vier servo's. Met behulp van drie markers kan de waarde van de ontvangen pulsen snel worden geschat.



figuur 8. Het is niet al te ingewikkeld, want het bestaat voor een groot deel uit connectoren. Het hart van de schakeling is een ATmega328P micro-controller, dezelfde als op de Arduino UNO. Zijn klokfrequentie wordt bepaald door X1, een 16MHz-kwarts kristal, met ondersteuning van C5 en C6.

Vier van zijn GPIO-poorten (PB0...PB3) zijn verbonden met de servo-sig-naalingangen op K1. De servo-sig-naaluitgangen (PD5, PD6, PD7 en PB4) zijn aangesloten op K2. Deze twee connectoren zijn zo bedraad dat standaard servokabels rechtstreeks kunnen worden aangesloten. Met andere woorden, de pintripletten 1-3-5, 2-4-6, 7-9-11

Figuur 8. Bij de superservotester is het ATmega328P-brein omringd door een groot aantal connectoren.



220069-001



en 8-10-12 corresponderen alle met een servo-aansluiting. Voor deze aansluitingen wordt vaak een oranje-rood-bruine kleurcode gebruikt. Oranje (pulssignaal) wordt aangesloten op pin 1 (of 2 of 7 of 8), rood (V_{CC}) op pin 3 (of 4 of 9 of 10) en bruin (GND) op pin 5 (of 6 of 11 of 12). Natuurlijk zijn er uitzonderingen op deze regel, dus controleer dat eerst voordat je iets aansluit.

Analoge ingangen

De vier potmeters zijn verbonden met de analoge ingangen van de MCU op PC0...PC3. De voedingsspanning wordt via een spanningsdelers (R_4 , R_5 en R_6) aangesloten op analoge ingang PC4. De verhouding tussen (R_4+R_5) en R_6 moet 2:1 bedragen, maar de absolute waarden zijn niet kritisch. Het gebruik van drie weerstanden van dezelfde waarde vereenvoudigt de selectie van zo goed mogelijk gelijke exemplaren.

Om de voedingsspanning van de microcontroller te meten is een andere referentiespanning dan de voedingsspanning zelf nodig voor de analoog/digitaal-omzetter (ADC). De ATmega328P heeft een ingebouwde referentie van 1,1 V, maar die waarde is wat laag. Ik heb daarom D2, een LM385-2,5, gebruikt als externe 2,5V-referentie. Deze component is nauwkeuriger dan een eenvoudige zenerdiode met twee aansluitingen, wat de kwaliteit van de meting van de servo-voedingsspanning ten goede komt: 1...2% in plaats van ongeveer 5% voor een gewone zenerdiode.

Merk op dat D2 wordt geleverd in een hoge behuizing. Om de tester compact en laag te houden, kan hij naar de printplaat worden gebogen. Aan de andere kant is het ook mogelijk om hem als steun voor het display te gebruiken. De te meten de voedingsspanning mag niet hoger zijn dan 7,49 V, aangezien de maximale ingangsspanning van de MCU 5 V bedraagt. Daarom mag de voeding van de ontvanger en de servo's deze waarde nooit overschrijden.

Display-problemen

Het op de SSD1306 gebaseerde I²C OLED-display is aangesloten op connector K4. De I²C-poort van het display is niet verbonden met de I²C-poort van de MCU, maar met PD0 en PD1. De I²C-bus wordt in software geëmuleerd. De reden is dat op de ATmega328 de I²C-bus gedeeld wordt met analoge ingang PC4, die al gebruikt wordt voor het meten van de voedingsspanning. Daarom kan de ingebouwde I²C-periferie hier niet worden gebruikt. R_9 en R_{10} zijn pull-ups voor de I²C-bus. Officieel zouden ze een waarde moeten hebben van om en nabij de 4,7 k Ω , maar 10 k Ω werkt ook en spaart een regel in de onderdelenlijst, omdat de meeste andere weerstanden ook een waarde van 10 k Ω hebben.

Merk op dat K4 is getekend als een tweerijige 8-polige connector, maar op de PCB moet je een eenrijige 4-polige connector monteren in positie 'A' of 'B', niet allebei. De positie hangt af van je display. Deze displays hebben de GND- en VCC-pinnen op pin 3 en 4, maar niet altijd in

dezelfde volgorde. Met K4A en K4B kun je beide typen gebruiken. Het gebruik van een dubbele footprint in plaats van (soldeer)jumper kost minder printruimte en bespaart twee (soldeer)jumper. Het nadeel is natuurlijk dat de positie van de displayopening in de behuizing afhankelijk is van het display, omdat K4A en K4B niet op dezelfde plaats zitten. Ook kunnen de afmetingen van dit display per fabrikant verschillen. Daarom moet je je display kiezen voordat je de behuizing op maat gaat maken.

Keuze van de bedrijfsmodus

Schuifschakelaar S1, een dubbelpolig tweestanden-exemplaar (DPDT), selecteert de bedrijfsmodus van de super-servotester. In de *manual*-modus wordt de 5V-voedingsspanning naar de servo-connectoren doorgeschakeld. In de *inputs*-modus is de voedingsspanning van de SST V_{BATT} , dus zijn eigen 5V-voeding moet worden losgekoppeld om conflicten te voorkomen. Eigenlijk zou je de voeding van de SST in deze modus zelf moeten uitschakelen, maar om het zekere voor het onzekere te nemen, doet S1 dat voor je. Zenerdiode D3 zorgt er samen met R_8 voor dat de voeding voor de rest van de schakeling (afkomstig van V_{BATT}) de maximale waarden die voor de andere componenten gelden, niet overschrijdt. De positie van S1 wordt gelezen door PD4, zodat de MCU de corresponderende bedrijfsmodus kan kiezen.

Strikt genomen hoeft S1 geen schuifschakelaar te zijn, maar hij moet wel in staat zijn het door maximaal vier servo's opgenomen vermogen te verwerken en zelfs meer als er ook een ESC of flight controller op de uitgangen is aangesloten. Geschikte DPDT-schuifschakelaars zijn vaak goedkoper dan dubbelpolige wipschakelaars, wat verklaart waarom we hier een schuifschakelaar gebruiken.

Diversen

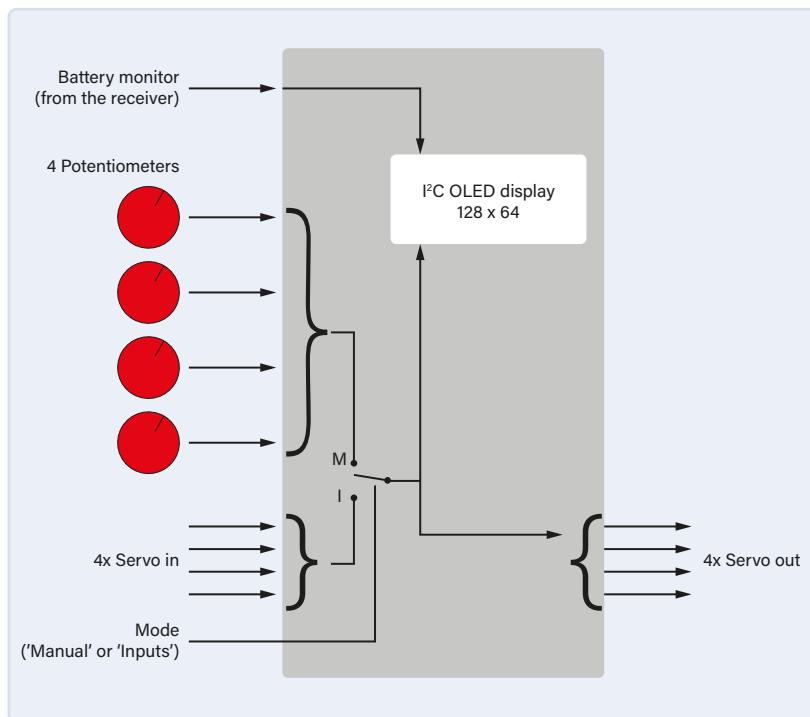
GPIO-poort PD2 van de MCU bestuurt de alarm-LED en de zoemer. Omdat deze parallel zijn geschakeld, zorgt MOSFET T1 voor wat extra puf om ze allebei aan te sturen zonder de MCU te zwaar te belasten. Met schuifschakelaar S2 kan de zoemer worden uitgeschakeld, omdat diens wat geluid soms irritant kan zijn. Deze schakelaar is een klein type voor geringe vermogens.

De 5VDC-voeding voor de SST in manual-modus wordt verkregen uit een klassieke lineaire spanningsregelaar van het type 7805 (IC1). Je moet hiervoor een TO220-type gebruiken, want hij moet maximaal vier servo's van stroom voorzien. Om de tester zo laag mogelijk te houden, kunnen C1 en C2 liggend worden gemonteerd.

Tenslotte is connector K3, een 'boxed' type, beschikbaar voor het in-circuit programmeren van de microcontroller. Hij is op dezelfde manier bedraad als de Arduino ICSP-connector.

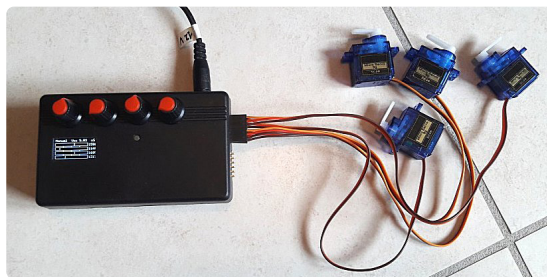
Enkele woorden over de software

Voor dit project heb ik de Arduino IDE gebruikt, wat bewijst dat deze ook voor complexe projecten geschikt is.

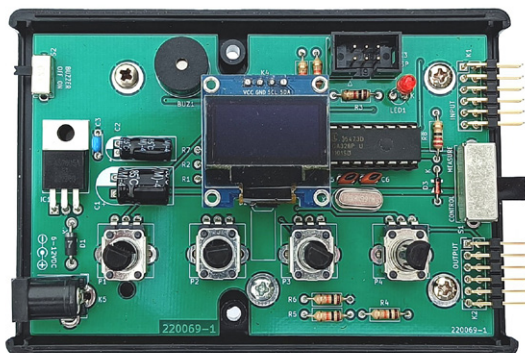


Figuur 9. High-level overzicht van wat er in de microcontroller gebeurt.

Figuur 10. Het prototype van de auteur bestuurt vier servo's in manual modus.



Figuur 11. Het Elektor Labs-prototype in zijn behuizing gemonteerd.



Ik bouwde eerst een proof-of-concept met een Arduino UNO op breadboard. Toen alles naar behoren werkte, laadde ik het programma in het definitieve apparaat via de ICSP-connector. Merk op dat bij het programmeren van de microcontroller via de ICSP-connector K3, niets mag worden aangesloten op connectoren K1 en K2, aangezien deze connectoren elk een signaal delen met K3. Alle voor het project benodigde bibliotheken zijn opgenomen in de download [3], behalve de servo-bibliotheek, omdat die deel uitmaakt van de Arduino IDE. Het project is georganiseerd in drie bestanden: *Display.ino*, *Inputs.ino* en het hoofd-sketchbestand. De namen van de bestanden maken vrij goed duidelijk wat ze bevatten. Het hoofd-sketchbestand bevat de functies `setup()` en `loop()`, maar ook de interrupt-routines die worden gebruikt om de ingangspulsen te timen. Niveauveranderingen in de ingangssignalen veroorzaken pin-change interrupts. Zo kan de functie `micros()` worden gebruikt om de lengte van de pulsen te meten. De gemeten pulsbreedtes worden gekopieerd naar de uitgang en doorgegeven aan de displayfuncties voor weergave. **Figuur 9** geeft een high-level overzicht van de signaalflow in de microcontroller.

Gebruik van de OLED_I2C bibliotheek

Zoals eerder opgemerkt, wordt het OLED-display aangestuurd via een I²C-bus. Op de ATmega328P wordt deze bus gedeeld met de ADC. PC4 kan zowel de SDA-pin als de analoge ingang A4 zijn. Om dit te omzeilen wordt de I²C-bus in software geëmuleerd. De OLED_I2C bibliotheek van Rinky-Dink Electronics [2] doet dit en stelt ons in staat elke GPIO-poort voor de I²C-bus te gebruiken. Ik moest het bestand *OLED_I2C.cpp* van deze bibliotheek enigszins wijzigen omdat een onjuiste `include`-opdracht de compilatie ervan verhinderde. Vervang regel 25:

```
#include "hardware/avr/HW_AVR.h"
```

door

```
#include "HW_AVR.h"
```

Ook heb ik in het bestand *HW_AVR.h*, in de functie `OLED::update()`, regel 24 uitgecommentarieerd, zoals hieronder:

```
//noInterrupts();
```

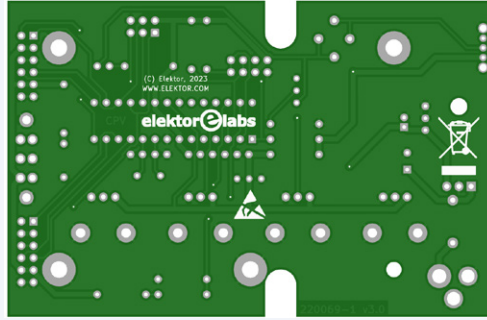
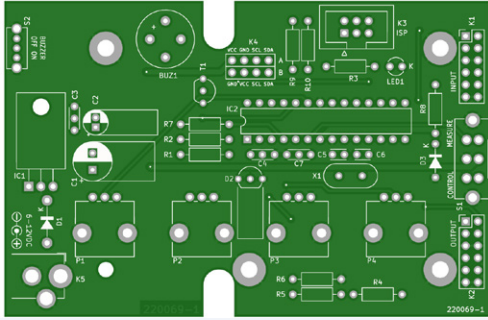
Dit is nodig omdat interrupts door de Servo-bibliotheek worden gebruikt voor het genereren van de 50-Hz PWM-signalen op de uitgangen van de SST. Ze moeten ook ingeschakeld blijven voor het meten van binnenkomende pulsen.

Voor het inschakelen

Als potentiometer P1 bij het inschakelen in de minimumstand staat (naar links gedraaid), geeft het scherm een wit



ONDERDELENLIJST



Weerstanden (THT, 0,25 W, 5%):

R1,R3 = 1 k
R2,R4,R5,R6,R7,R9, R10 = 10 k
R8 = 22 Ω
P1,P2,P3, 4 = 10 k lineair, verticaal

Condensatoren (THT):

C1 = 100 μ /35 V, 8 mm diam., steek 3,5 mm
C2 = 10 μ /16 V, 5 mm diam., steek 2 mm
C3,C4,C7 = 100 n, steek 2,5 mm of 5 mm
C5,C6 = 22 p, steek 2,5 mm

Halfgeleiders (THT):

D1 = 1N5817
D2 = LM385Z-2.5, TO-92
D3 = BZX79-C5V1
IC1 = 7805, TO220
IC2 = ATmega328-P
LED1 = LED, 3 mm, rood
T1 = 2N7000, TO-92

Diversen:

BUZ1 = Piëzo-zoemer met oscillator, 12 mm diam, steek 6,5 mm of 7,6 mm
K1,K2 = 2-rijige, 12-polige pinheader, raster 2,54 mm, haaks
K5 = voedingsconnector
K4* = 4-polige busstrip, raster 2,54 mm (monteer A of B afhankelijk van display)
K3 = 2-rijige 6-polige boxed pinheader, raster 2,54 mm
S1 = schuifschakelaar DPDT, haaks (bijv. MFS201N-16-Z)
S2 = schuifschakelaar SPDT, haaks (bijv. OS102011MA1)
X1 = kristal, 16 MHz, HC-49S (laag profiel)

OLED-display, 0,96", 128 x 32 pixels, 4-polig I2C
(bijv. www.elektor.nl/18747)

Elektor PCB 220069-1

aanbevolen behuizing: Hammond 1593N

* = zie tekst

kader weer. Dit helpt om de uitsparing voor het scherm te positioneren en maakt het mogelijk om de opening in de behuizing zo nodig aan te passen. Niet iedereen heeft thuis een CNC-machine. Draai P1 naar het midden om terug te keren naar de normale werking.

Als P4 bij het inschakelen in de minimumstand staat, gebruikt het display de *stack*-modus. Als P4 bij het inschakelen in de maximale stand staat, wordt de *square* displaymodus geactiveerd. Dit laatste geval houdt in dat de op uitgang 4 aangesloten motor bij het opstarten op volle snelheid draait.

Tijd om op te stijgen

De super-servotester was leuk om te ontwerpen en te ontwikkelen en is een nuttig instrument om in de buurt te hebben als je je bezighoudt met drones en quadcopters of RC-modellen in het algemeen. Dit project heeft me niet alleen geholpen om mijn programmeervaardigheden te

vergroten, maar ik kan nu ook eindelijk mijn drone tevoorschijn halen, die ongeduldig wacht om op te stijgen! 

220069-03

Vragen of opmerkingen?

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



Gerelateerde Producten

- > **Arduino Uno R3 (SKU 15877)**
www.elektor.nl/15877
- > **0,96" 128x64 I2C OLED Display (SKU 18747)**
www.elektor.nl/18747

WEBLINKS

- [1] Joop Brokking's drone-website: http://www.brokking.net/ymfc-al_main.html
- [2] OLED_I2C bibliotheek: <http://www.rinkydinkelectronics.com/index.php>
- [3] Project-downloads op Elektor Labs: <https://www.elektormagazine.com/labs/super-servo-tester>



Analoge signalen en microcontrollers

ADC's, DAC's, stroommeting en meer

Mathias Claussen (Elektor)

Terwijl een MCU intern met enen en nullen werkt, is de wereld eromheen analoog. Daarom is het soms nodig om analoge waarden in te lezen of uit te voeren. Met slechts een paar componenten kan een MCU worden aangesloten op de analoge wereld.

Of je microcontroller (MCU) nu een Arduino UNO, STM32 of wat dan ook is – vaak is het nodig om meer dan alleen digitale signalen te verwerken. De wereld om ons heen is analoog: niet alleen licht of donker maar ook elke tint daartussen. Daarom wil dit artikel enig inzicht geven in hoe analoge waarden kunnen worden ingelezen door een microcontroller en hoe ze ook kunnen worden uitgevoerd. Voor ervaren ontwikkelaars valt deze tekst zeker in de categorie 'lichte lectuur', maar we moesten allemaal ooit ergens beginnen. En hoewel het instellen en lezen van digitale pinnen heel eenvoudig is, wordt het een beetje lastiger als het gaat om het inlezen en uitvoeren van analoge waarden.

De pricipiële vragen zijn: hoe krijgen we analoge waarden in een digitaal werkende MCU, en hoe moeten ze daartoe worden voorbereid? En hoe kan een MCU analoge waarden uitvoeren? Om deze vragen te beantwoorden kijken we eerst naar twee belangrijke basiselementen, de analoog/digitaal-converter (ADC) en de digitaal/analoog-converter (DAC).

Analoog/digitaal-converter (ADC)

Een analoog/digitaal-converter (ADC) kan een analoog signaal omzetten in een digitale representatie. Deze kan dan later door de microcontroller worden gebruikt om de gegevens te verwerken. Twee veel voorkomende parameters van een ADC zijn de resolutie in bits en de bemonsteringssnelheid in monsters (samples) per seconde. Bij microcontrollers ligt de resolutie meestal tussen 10 en 12 bit. Dit betekent dat de ADC zijn analogeingangsspannings-

bereik onderverdeelt in 1024 waarden (10 bit) of 4096 waarden (12 bit) en vervolgens de aangelegde spanning toewijst aan een van deze waarden. Het waardenbereik is nu eindig en het analoge signaal moet worden toegewezen aan de best passende digitale waarde, ook al komt die niet precies overeen met het rekenkundig bepaalde niveau. Deze afwijking wordt kwantisatiefout genoemd. De tweede grootte, de bemonsteringsfrequentie in samples per seconde, geeft aan hoeveel keer per seconde de ADC een analoge waarde kan omzetten in een digitale waarde. Tenzij het signaal een gelijkspanning is, bepaalt de bemonsteringsfrequentie ook de maximale bandbreedte van een signaal dat kan worden verwerkt. Het omgekeerde van de bemonsteringsfrequentie geeft aan hoe lang het duurt voordat de ADC een analoog/digitaal-conversie heeft voltooid – met andere woorden, hoe lang de software moet wachten nadat een conversie is gestart voordat de resulterende waarde kan worden verwerkt.

Spanningsbereik, resolutie en referentiespanning

Als we uitgaan van een ingangsspanningsbereik van 0 V tot 5 V met een resolutie van 10 bit, krijgen we:

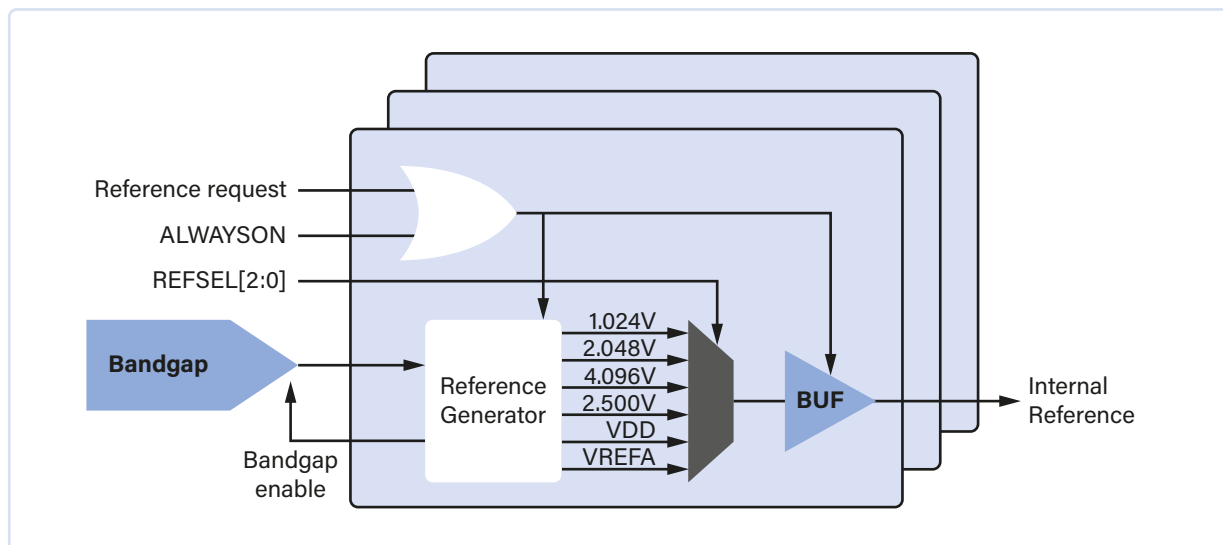
$$5 \text{ V} / (2^{10} \text{ bit} - 1) = 4,89 \text{ mV/bit}$$

Een spanning van bijvoorbeeld 452 mV krijgt zodoende intern de waarde 92:

$$452 \text{ mV} / 4,89 \text{ mV/bit} = 92$$

en zou dus verder verwerkt worden als 449,88 mV. De kwantisatiefout bedraagt dus 2,12 mV. De afwijking kan niet groter zijn dan 0,5 LSB (minst significante bit), wat in dit voorbeeld net geen 2,5 mV is.

Als het te meten spanningsbereik aanzienlijk kleiner is dan het toegestane ingangsspanningsbereik van de ADC, kan dit bereik bij de meeste ADC's worden aangepast. De ADC vergelijkt de analoge waarden altijd met een referentiespanning. Veel MCU's bieden tegenwoordig een interne referentiespanning, of zelfs meerdere. De AVR128DB van Microchip dient als voorbeeld. **Figuur 1** toont het blokschema van de spanningsreferentie van deze controller. Zes referentiespanningen van 1,024 V, 2,048 V, 4,096 V, 2,5 V, VDD en



Figuur 1. Spanningsreferentie AVR128DB (bron: Microchip, AVR DB Family datasheet [PDF] – <https://elektor.link/AVR128DBDatasheet>).

V_{REFA} zijn beschikbaar. De eerste drie spanningen kunnen worden voorgesteld als 2^N en maken het mogelijk de digitale waarden van de ADC later heel gemakkelijk terug te rekenen naar volt. Iets dat op bijna alle MCU's te vinden is, is GND (massa) versus AGND (massa voor het analoge deel van de MCU). Omdat er in het digitale domein hoogfrequente ruis kan optreden die ook door het massapad loopt, heeft alles wat een ADC of DAC is zijn eigen massa. In het ontwerp kunnen GND en AGND dus van elkaar worden ontkoppeld om storingen te voorkomen. Indien verwacht wordt dat de gemeten grootheid niet groter zal zijn dan 1 V, kan 1,024 V als referentie worden gekozen. Dit zou dan resulteren in een resolutie van 1,00098 mV/bit. 452 mV zou dan overeenkomen met de digitale waarde 452; bij de verdere verwerking zou de kwantisatiefout dus aanzienlijk kleiner zijn dan in het eerste voorbeeld. De juiste keuze van de referentiespanning is echter niet het enige dat ervoor kan zorgen dat de resultaten van de conversies dicht bij de fysieke (analoge) waarde liggen. Ook de kwaliteit van de referentiespanning is van invloed.

In de AVR128DB kunnen VDD en V_{REFA} ook als referentiespanningen worden beschouwd. VDD is de voedingsspanning van de chip, en V_{REFA} is een externe pin van de AVR128DB waarop je je eigen externe referentiespanningsbron kunt aansluiten. Het gebruik van VDD als referentie kan een ongelukkige keuze zijn, omdat deze spanning belast is met ruis van het digitale deel van de MCU, en het is moeilijk deze ruis 'weg te filteren'. De situatie is anders met V_{REFA} : hier kan een externe hoge-precisie spanningsbron worden gebruikt als referentie. Natuurlijk is deze nog steeds verbonden met de 'vervuilde' voedingsspanning, maar als externe component kan deze veel beter worden gefilterd. Een stabiele en storingsvrije referentie is de basis van een goede analoog/digitaal-conversie. Er is nog een andere manier om de resolutie van een ADC te verhogen zonder de referentiespanning te veranderen; dit wordt oversampling genoemd. Oversampling gebruikt het minst significante bit, dat heen en weer geschakeld wordt tussen nul en één als de signaalwaarde niet precies overeenkomt met de beschikbare resolutie

van de ADC. Door het gemiddelde van dit minst significante bit te nemen wordt een fractie van een één verkregen, wat een hogere resolutie betekent. Deze methode verlaagt echter de effectieve bemonsteringssnelheid. Voor elk bit dat door oversampling wordt toegevoegd, wordt de bemonsteringssnelheid met een factor 4 gereduceerd. Als bijvoorbeeld een 12bit-ADC maximaal 130 ksp/s (kiloamples per seconde) aankan, zouden vier oversampling-bits, dus 16 bit als totale resolutie, resulteren in slechts 507 samples per seconde:

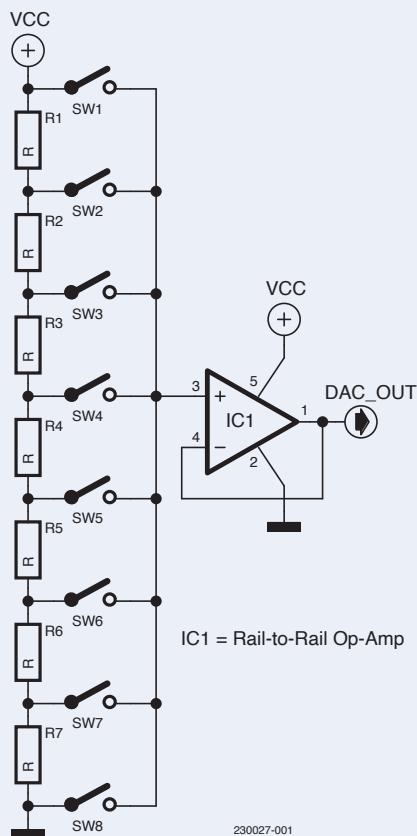
$$130 \text{ ksp/s} / 4^4$$

Bovendien moet de processor, afhankelijk van de ADC, het gemiddelde berekenen (nodig voor oversampling) wat extra rekenwerk oplevert. Sommige nieuwere ADC's, zoals die van sommige STM32-controllerfamilies, voeren deze taak volledig autonoom uit zonder de CPU extra te belasten.

ADC's zijn er in verschillende uitvoeringen. De meest voorkomende ADC's in controllers zijn van het SAR-type (successive approximation). Voor een hoge resolutie (24 bit en meer) bij relatief lage samplesnelheden (van ksp/s tot enkele samples per seconde) worden meestal delta/sigma-ADC's gebruikt. De details van de delta/sigma-conversie en de implementatie ervan in een CPLD/FPGA (Complex Programmable Logic Device/Field-Programmable Gate Array) zijn te vinden in een artikel van Elektor [1]. Er zijn andere ADC-types, waarvan het Flash ADC-type de snelste conversietijd belooft, en daarom vaak gebruikt wordt in modern flash-geheugen.

Digitaal/analoog-converter (DAC)

Met een DAC genereer je een analoge waarde uit een digitale waarde. In sommige MCU's is een DAC ingebouwd, hoewel niet zo vaak als een ADC. Bij de AVR128DB is dat een 10bit-DAC, die dus 1024 verschillende spanningen kan produceren. Als er geen DAC in de MCU zit, kan een externe DAC-chip zoals de Microchip MCP4922 worden gebruikt. In beide gevallen betreft het een DAC met een



Figuur 2. Structuur van een DAC met weerstandsketen.

weerstandsketen, waarbij de MCP4922 is een tweekanaals 12bit-DAC is. Terwijl een weerstandsketen-DAC goed kan worden geïntegreerd in een chip, is zoiets nauwelijks haalbaar met discrete weerstanden. Een blik op de 3-bit weerstandsketen-DAC van **figuur 2** maakt aannemelijk waarom dat zo is: voor de DAC zijn

$$2^N - 1 \quad (N = \text{aantal bits})$$

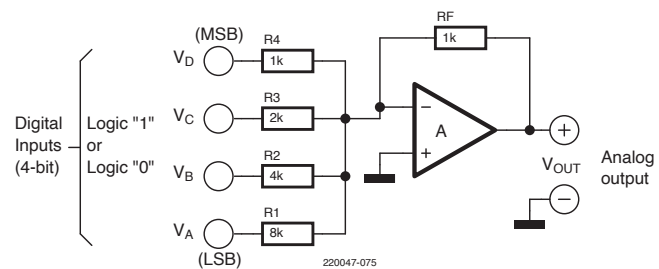
weerstanden en N schakelaars nodig. Voor een 16bit-DAC betekent dat dus 65.535 weerstanden en 65.536 schakelaars! Net als een ADC heeft ook een DAC een referentiespanning nodig die het bereik aangeeft waarbinnen de uitgangsspanning kan liggen. De DAC zelf kan echter slechts een paar mA aansturen, zelfs als de uitgang wordt gebufferd door een opamp.

Als er echter geen interne DAC is en er nog voldoende pinnen van de MCU beschikbaar zijn, bestaan er andere manieren om een externe DAC te bouwen. Eén optie is een binair gewogen DAC, zoals in **figuur 3**. Hier zijn slechts N+1 weerstanden en een opamp nodig voor N bits. De waarden van de weerstanden worden gegeven door

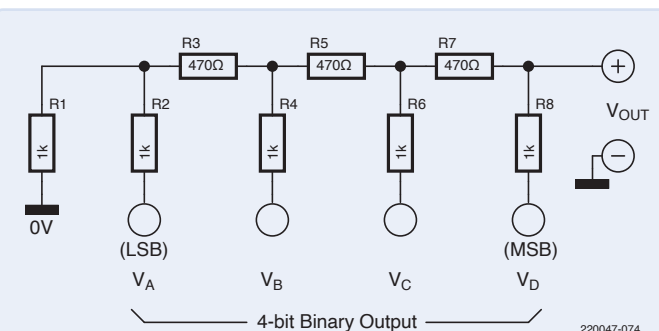
$$R_N = R_{(N-1)} \cdot 2$$

en worden dus telkens verdubbeld. Een dergelijke DAC kan worden gebruikt om bijvoorbeeld analoge VGA-signalen te genereren, zoals in de datasheet bij [2].

Een andere veelgebruikte variant is een R-2R DAC. In tegenstelling tot een binair gewogen DAC zijn hier 2N weerstanden en een opamp nodig voor N bits, maar er zijn slechts twee weerstandswaarden, terwijl we in een binair gewogen DAC met N meestal lastig



Figuur 3. Schakeling van een binair gewogen DAC.

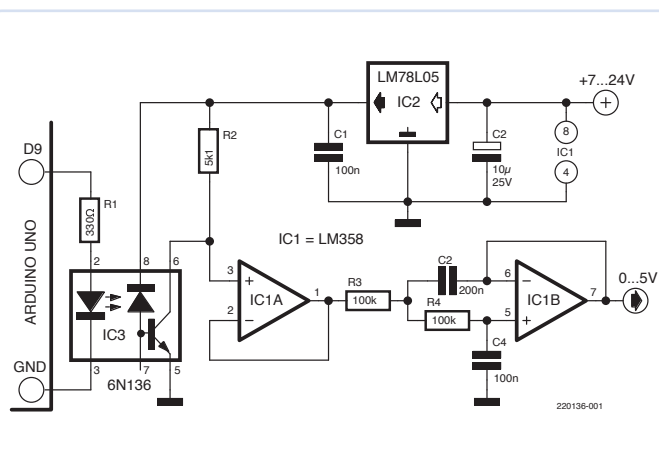


Figuur 4. Principe van een R-2R DAC.

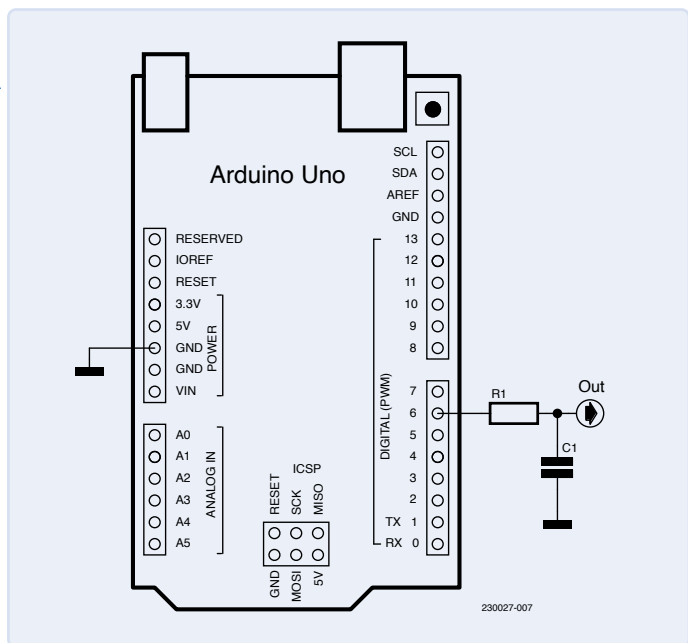
te vinden fractionele waarden te maken hebben. Het schema van zo'n DAC is geschetst in **figuur 4**.

Als I/O-pinnen echter schaars zijn op de gebruikte MCU, kan de PWM-eenheid met wat externe onderdelen ook als DAC dienen. De `analogWrite()` functie voor de Arduino UNO produceert op deze manier analoge waarden met behulp van de digitale pinnen. **Figuur 5** toont een voorbeeld uit het Elektor-artikel in [3] voor een Arduino UNO.

Een eenvoudiger opzet is de schakeling van **figuur 6**, die alleen bestaat uit een RC-laagdoorlaatfilter om het signaal af te vlakken. Een dergelijke DAC heeft echter nadelen. Aangezien het PWM-sig-naal moet worden afgevlakt, moeten de waarden voor R en C



Figuur 5. Geïsoleerde analoge uitgang voor de Arduino UNO.



Figuur 6. Arduino UNO met RC-laagdoorlaatfilter.

worden afgestemd op de betreffende toepassing. Ook zal er altijd een klein PWM-residu in het analoge signaal achterblijven, dat als ongewenste ruis merkbaar wordt.

Signaalvoorbereiding

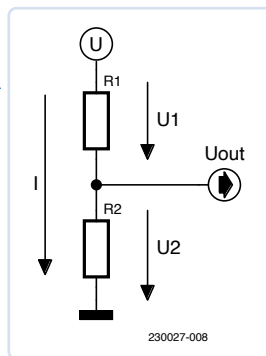
De te meten of uit te voeren signalen zijn niet altijd direct compatibel met de gekozen MCU. In de digitale wereld zijn veel ontwikkelaars bekend met het probleem van het aansluiten van 5V-componenten op 3,3V-systemen. Soortgelijke problemen doen zich ook voor in de analoge wereld en diens interfaces. Audio, video en 4-20 mA current loops zijn slechts enkele voorbeelden die signaalvoorbereiding (signaalconditionering) vereisen. De eenvoudigste vorm van signaalconditionering is waarschijnlijk een spanningsdeler, die niet alleen kan worden gebruikt voor analoge waarden, maar ook voor het aanpassen van digitale signalen.

Spanningsdelers en ADC's

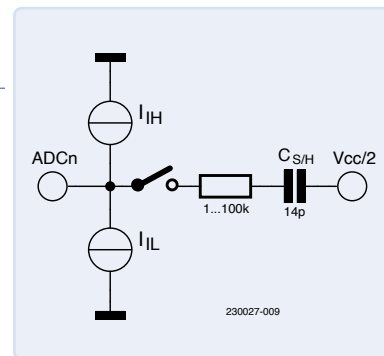
Figuur 7 toont een onbelaste spanningsdeler – dat is de geïdealiseerde versie. Wanneer een spanning (U) wordt aangelegd, gaat er een stroom (I) door $R_1 + R_2$ lopen. Over weerstand R_1 valt nu een spanning van $R_1 \times I = U_1$ en over R_2 een spanning $R_2 \times I = U_2$, waarbij $U = U_1 + U_2$. Als je nu bijvoorbeeld een bereik van 0 tot 10 V wilt meten met een ADC die alleen 0 tot 5 V kan verwerken, moet de spanning gehalveerd worden. Het is duidelijk dat dan $R_1 = R_2$ moet zijn.

De stroom (I) moet echter ook in aanmerking worden genomen bij de keuze van de weerstandswaarden. Hoewel 10Ω voor R_1 en R_2 rekenkundig gezien zou resulteren in de gewenste spanningen voor U_1 en U_2 , zou er een stroom van 500 mA door de weerstanden lopen. Dit zou niet alleen de spanningsbron kunnen overbelasten, maar ook de weerstanden zelf. Een vermogensdissipatie van $P = R \times I^2 = 2,5 \text{ W}$, die elke weerstand in de vorm van warmte moet dissiperen, is namelijk veel te hoog voor de meeste kleinere weerstanden.

Een andere optie om de vermogensdissipatie te beperken, zijn weerstanden van $1 \text{ M}\Omega$. Dan zou er een stroom van slechts $5 \mu\text{A}$ lopen, en de vermogensdissipatie zou $25 \mu\text{W}$ bedragen, maar dit andere uiterste veroorzaakt een nieuw probleem. Als in de ADC



Figuur 7. Onbelaste spanningsdeler.

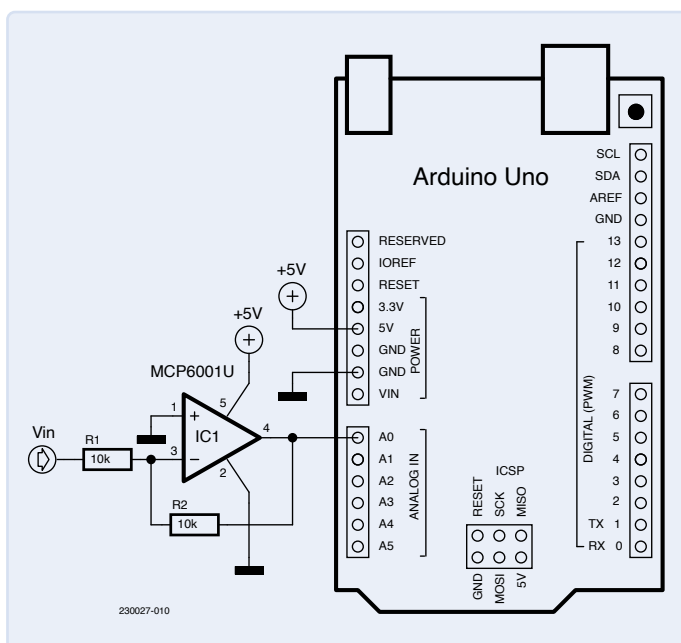


Figuur 8. ATmega328P ADC interne ingangsstructuur (bron: Microchip, Atmel Atmega328P datasheet [PDF] – <https://elektor.link/ATmega328PDatasheet>).

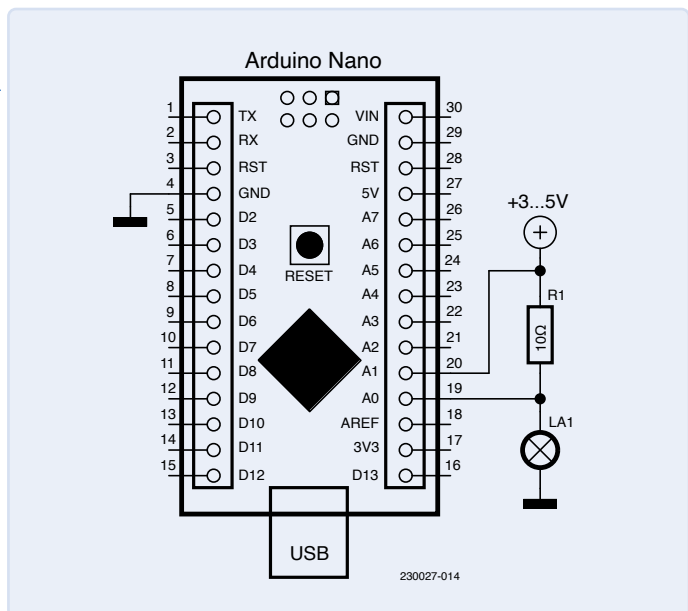
een sample&hold-trap is ingebouwd, zoals in de ATmega328 in **figuur 8**, kan dit leiden tot foutieve meetwaarden. De ADC heeft een kleine condensator aan de ingang die opgeladen wordt. Tijdens een conversie wordt de condensator voor een korte tijd (voor de duur van de ingestelde bemonsteringstijd) verbonden met de analoge ingang en opgeladen. Als de stroom te gering is, bestaat de mogelijkheid dat de condensator tijdens de bemonsteringstijd niet volledig wordt geladen. Als de ADC later de waarde van de onvolledig geladen condensator omzet, zal het resultaat onjuist zijn. Een blik op de datasheet van de MCU van je keuze helpt om de juiste waarden te bepalen. Voor de AVR in dit geval zou een waarde van $4,7 \text{ k}\Omega$ voor R_1 en voor R_2 en dus een stroom van $1,06 \text{ mA}$ de juiste keuze zijn.

Negatieve spanningen

Positieve spanningen zijn vrij gemakkelijk te meten, bijvoorbeeld met een spanningsdeler. Maar wat gebeurt er als de te meten spanning negatief is? Door een opamp met een paar weerstanden te gebruiken als inverterende versterker, zoals in **figuur 9**, kan de spanning worden geïnverteerd.



Figuur 9. Inverterende versterker.



Figuur 13. High-side meting voor en na de shuntweerstand.

De shuntweerstand wordt tussen de positieve voedingsspanning en de belasting opgenomen. Om de stroom door de weerstand en dus door de belasting te bepalen, moet de spanningsval over de weerstand worden gemeten. Indien de (constante en bekende) voedingsspanning van de belasting kleiner is dan of gelijk aan de referentiespanning van de ADC, kan de spanningsval over de shunt relatief eenvoudig worden bepaald. Als de voedingsspanning van de belasting echter varieert is (maar nog steeds lager dan de referentiespanning van de ADC), moeten de spanningen aan beide zijden van de shuntweerstand worden gemeten, zoals geschetst in **figuur 13**.

Toegegeven, dit voorbeeld is erg geïdealiseerd omdat de voedingsspanning van de belasting kleiner is dan of gelijk is aan de referentiespanning van de ADC. Als de voedingsspanning groter is dan de referentiespanning van de ADC of zelfs van de MCU, wordt het moeilijker.

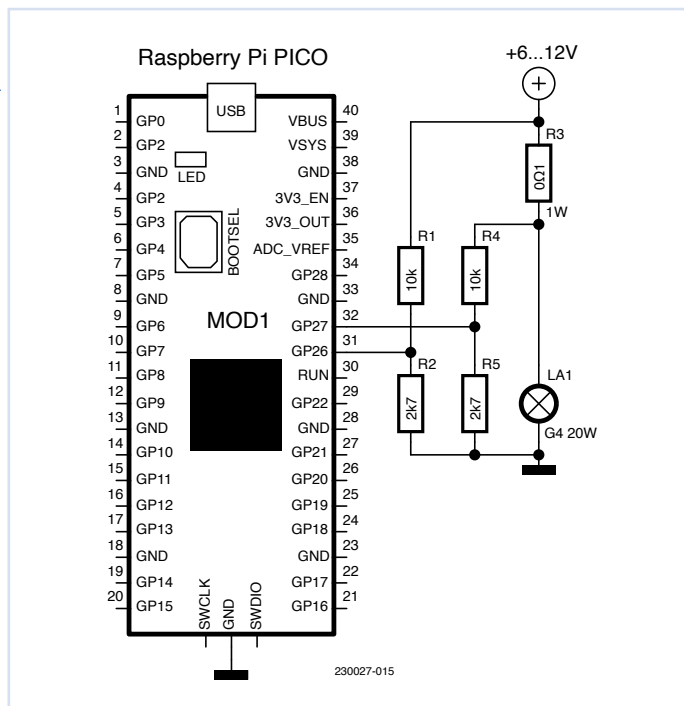
De stroom door een halogeenlamp meten

In het volgende voorbeeld willen we het stroomverbruik van een dimbare G4-halogeenlamp meten en monitoren, die wordt gevoed met een maximale gelijkspanning van 12 V. De lamp mag maximaal 20 W 'verstoken' (1,67 A bij 12 V). Een stroom van meer dan 2 A of minder dan 25 mA moeten door de MCU worden geïnterpreteerd als 'lamp defect'.

We zouden nu spanningsdelers kunnen gebruiken om de spanning voor en na de shuntweerstand in een bereik te brengen dat door de ADC kan worden verwerkt. **Figuur 14** toont een schakeling om dit voor elkaar te krijgen.

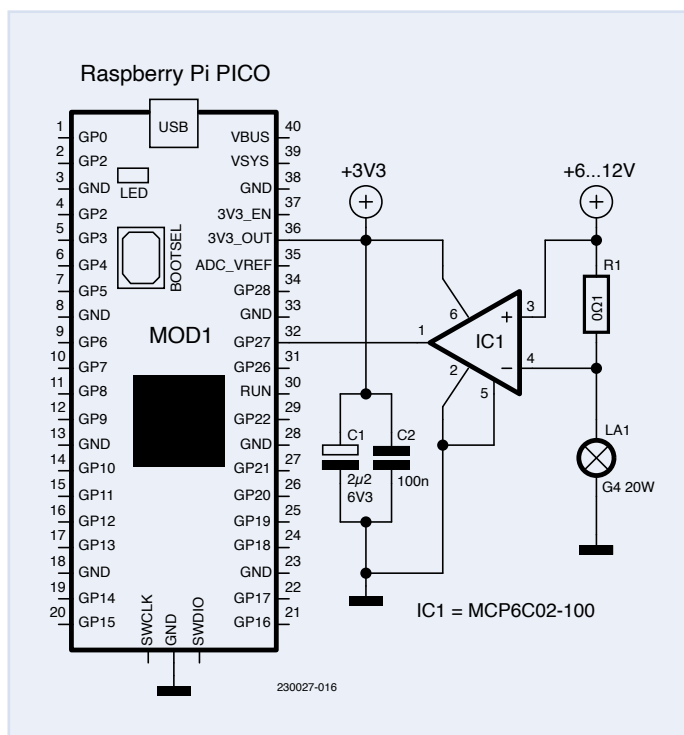
De MCU wordt gevoed met 3,3 V en de weerstanden zijn zo gekozen dat er bij 12 V maximaal 2,6 V op de pennen van de ADC komt te staan. De waarde van de shuntweerstand is 0,1 Ω. Bij 2 A veroorzaakt deze een spanningsval van 0,2 V, maar dissipeert nog steeds 0,4 W als warmte. De methode werkt in principe, maar is niet erg slim, zoals we straks zullen zien.

De spanningsdeler reduceert het meetbereik nog verder, zodat de ADC van de MCU een maximale spanningsval van minder dan 0,05 V bij 2 A moet kunnen verwerken. Ook het feit dat de spanningsval wordt gemeten via twee ADC-kanalen maakt de verdere verwerking niet eenvoudiger.

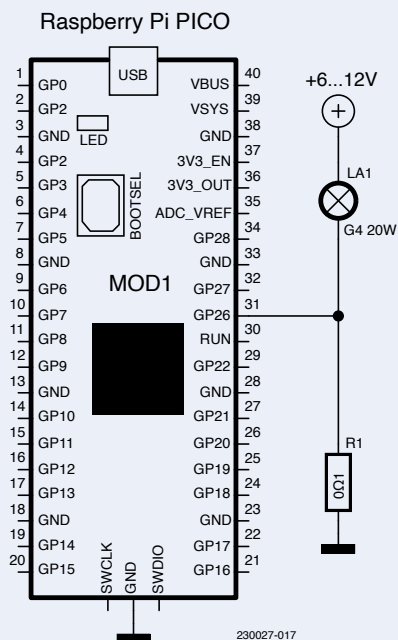


Figuur 14. Aangepaste schakeling met spanningsdelers.

In dit geval is het gebruik van een high-side stroommeetversterker zoals de INA138 van Texas Instruments of de MCP6Co2 van Microchip een veel betere oplossing. Bovendien kan de waarde van de shuntweerstand dan aanzienlijk kleiner zijn. Een shunt van 10 mΩ dissipeert veel minder dan zijn tien keer zo grote tegenhanger. **Figuur 15** toont de schakeling met stroommeetversterker. De uitgangsspanning is evenredig met de spanningsval over de weerstand en kan dus gemakkelijk worden verwerkt met een ADC-kanaal.



Figuur 15. Schakeling met stroommeetversterker.



Figuur 16. Low-side meting.

Low-side meting

In plaats van een shuntweerstand direct voor de belasting te plaatsen, kan deze ook worden aangesloten tussen de belasting en massa. Laten we bij het voorbeeld van de halogeenlamp blijven. **Figuur 16** toont zo'n low-side meting.

Hier wordt een weerstand van $0,1 \Omega$ gebruikt, die bij 2 A een spanningsval veroorzaakt van 0,2 V. Het voordeel is nu dat de potentiaal van de te meten spanningen onder die van de referentiespanning van de ADC en de voedingsspanning van de MCU ligt. Bij 25 mA bedraagt de spanningsval echter slechts 2,5 mV (0,1 mV per mA). Onder ideale omstandigheden zou de 12bit-ADC met een referentiespanning van 3,3 V nog steeds een resolutie van 0,8 mV per bit hebben. Er moet echter worden gekeken naar het effectieve aantal bits (Effective Number Of Bits, ENOB) van de ADC, rekening houdend met de meetfouten die in de praktijk optreden als gevolg van kwantisatiefouten, vervorming, ruis en dergelijke. Voor de RP2040 is een ENOB van ongeveer 9 gespecificeerd, wat neerkomt op ongeveer 6,44 mA per bit. Voor deze toepassing zou de resolutie van de stroommeting net voldoende zijn. Als echter een hogere resolutie nodig is, kan een opamp worden ingezet, waardoor dan ook een kleinere weerstandswaarde mogelijk is.

Opamps en temperatuursensoren

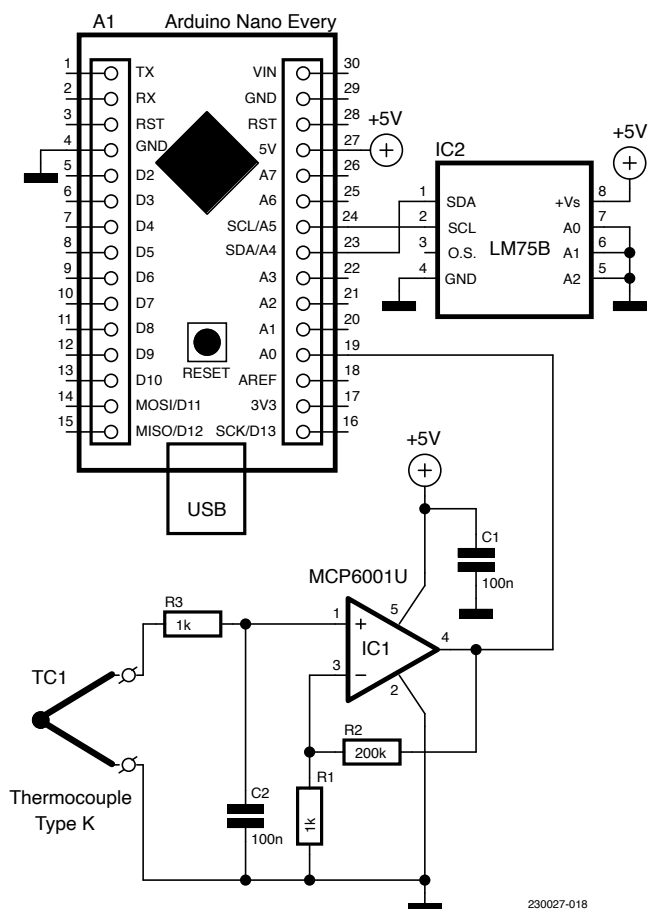
Hele bibliotheken zijn volgeschreven over operationele versterkers. Bij de juiste bedrading hebben deze fundamentele en veelzijdige componenten een talloze toepassingen, waarvan er in dit artikel al enkele zijn genoemd. Bij het meten van spanningen zijn het niet alleen shuntweerstand van waarvan de spanningen zich in een minder dan ideaal bereik bevinden voor de ADC van een MCU. Temperatuursensoren zijn ook kandidaten voor deze behandeling, tenminste als het zuiver analoge, resistieve types zijn zoals een PT100 of thermokoppels van twee verschillende metalen (in tegenstelling tot exemplaren met een digitale interface zoals de DS18B20 of de DHT11).

Thermokoppels

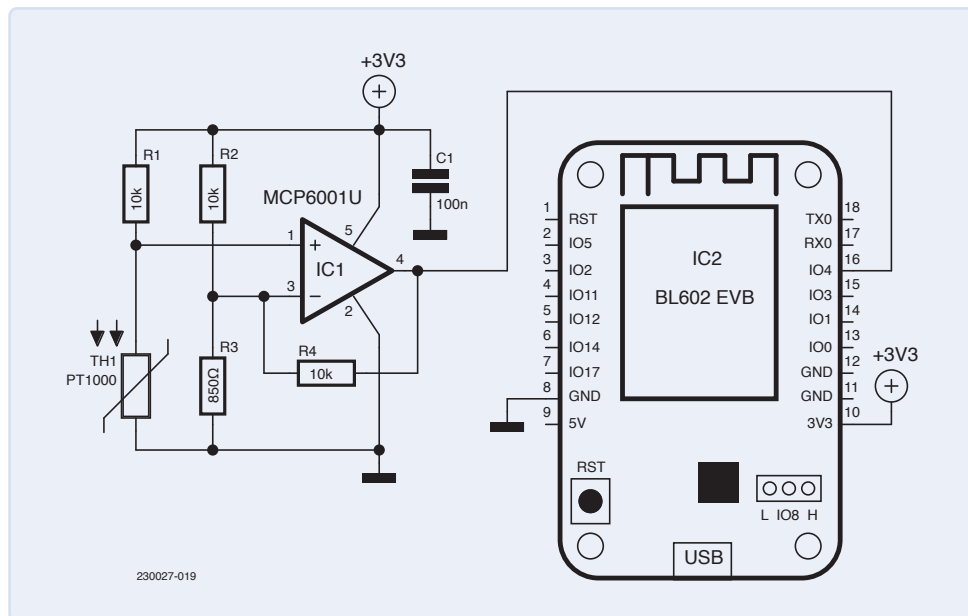
Deze uit twee verschillende metalen bestaande elementen fungeren als temperatuurafhankelijke spanningsbronnen. Het veelgebruikte K-type thermokoppel dient als voorbeeld. Bij 21°C of $69,8^\circ\text{F}$ levert zo'n component een spanning van 0,838 mV als gevolg van het Seebeck-effect; bij 22°C of $71,6^\circ\text{F}$ is dat 0,879 mV. Het verschil van slechts ongeveer $40 \mu\text{V}$ per graad celsius stelt de ADC van een MCU zwaar op de proef. Bovendien is bij gebruik van thermokoppels ook koude-las-compensatie nodig [4].

Een opamp kan worden gebruikt om de spanning van een thermokoppel te versterken. **Figuur 17** toont een voorbeeld van een dergelijke schakeling, opgezet voor een Arduino Nano Every, waarbij een temperatuurbereik van 0°C tot 400°C wordt verwacht (0...16,396 mV thermokoppel-spanning).

Hier wordt een niet-inverterende opamp gebruikt met een versterking $A = 200$. Om storingen aan de ingang van de operationele versterker te minimaliseren is een RC-filter toegevoegd; een eenvoudige digitale temperatuursensor van het type LM75 zorgt voor de temperatuurdetectie ten behoeve van de koude-las-compensatie.



Figuur 17. Uitlezen van een thermokoppel met de Arduino Nano Every.



Figuur 18. Brug van Wheatstone voor een PT1000.

PT100 en PT1000

PT100 en PT1000 zijn temperatuurafhankelijke weerstanden met een waarde van 100 Ω (PT100) of 1000 Ω (PT1000) bij 0 $^{\circ}\text{C}$. Om de temperatuur te berekenen moet eerst de weerstandswaarde van de sensor worden gemeten. Een goede mogelijkheid om die weerstandswaarde nauwkeurig te bepalen is de brug van Wheatstone. Net als bij thermokoppels zijn de spanningen bij PT100 en PT1000 vrij gering, en moet het kleine verschil tussen twee meetpunten worden bepaald. Ook hier biedt een opamp (aangesloten als niet-inverterende verschilversterker) de nodige assistentie. De schakeling voor een board met een voedingsspanning van 3,3 V, zoals het Pinecone BL602-evaluatieboard, is getekend in **figuur 18**.

Het spanningsverschil tussen de positieve en negatieve ingang van de opamp wordt versterkt. Maar wat als de temperatuur lager wordt dan 0 $^{\circ}\text{C}$? In dat geval zou de opamp een negatieve spanning moeten uitvoeren, wat niet mogelijk is. Een optie is hier om de weerstand R3 van 10 k Ω te vervangen door een weerstand van 850 Ω . De spanning tussen R1 en R2 zou dan 0,3917 V zijn, wat dan ook geldt voor het punt tussen R3 en de PT1000 bij een temperatuur van -38 $^{\circ}\text{C}$. Het meetvenster zou dus beginnen bij -38 $^{\circ}\text{C}$ met een spanningsverschil van 0 V. Bij 20 $^{\circ}\text{C}$ zou het spanningsverschil in de schakeling 0,09444 V bedragen.

Samenvattend

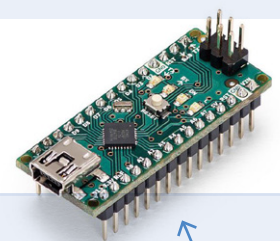
De verwerking en uitvoer van analoge waarden kan met acceptabele inspanningen plaatsvinden met een microcontroller. Ook al lijkt het proces niet zo eenvoudig als het aansturen van een digitale

I/O-pin, de omgang met de ADC is iets wat iedereen zou moeten proberen. De hier getoonde voorbeelden zijn bedoeld als suggesties voor eigen pogingen om de microcontroller te laten interageren met de analoge wereld buiten zijn behuizing. 

230027-03

Vragen of opmerkingen?

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



Gerelateerde producten

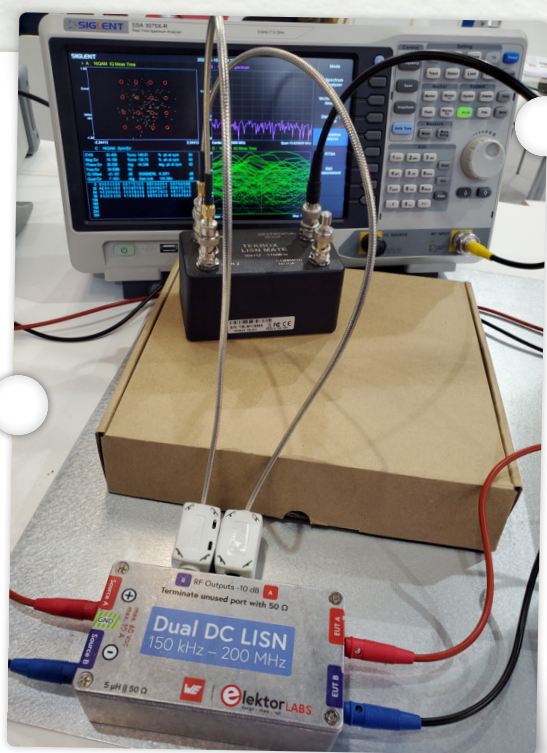
- > **Arduino Nano (SKU 17002)**
www.elektor.nl/17002
- > **Raspberry Pi Pico RP2040 (SKU 19562)**
www.elektor.nl/19562
- > **Pinecone BL602 Evaluation Board (SKU 19914)**
www.elektor.nl/19914

WEBLINKS

- [1] Guido Nopper, "Eenvoudige A/D-converter met PLD", Elektor september/oktober 2019:
<http://www.elektormagazine.nl/magazine/elektor-112/51093>
- [2] Hardware design with RP2040: <https://datasheets.raspberrypi.com/rp2040/hardware-design-with-rp2040.pdf>
- [3] Giovanni Carrera, "Geïsoleerde analoge uitgang voor Arduino Uno", Elektor november/december 2022:
<http://www.elektormagazine.nl/magazine/elektor-282/61132>
- [4] Thermokoppel (Wikipedia): <https://nl.wikipedia.org/wiki/Thermokoppel>

embedded world 2023

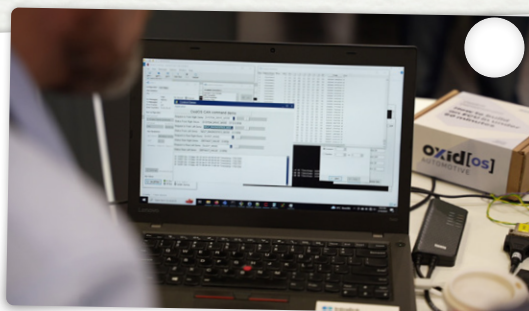
embedded world 2023 in Neurenberg – een must voor iedereen die met microcontrollers en tools te maken heeft – vond in de gebruikelijke periode in het voorjaar plaats. Dit jaar waren er 27.000 bezoekers, een stijging van 50% ten opzichte van de beurs van 2022, met meer dan 950 exposanten in 6 hallen. Elektor-redacteur Stuart Cording, Content Director CJ Abate en hoofdredacteur Jens Nickel hadden de kans om rond te kijken en een heleboel interessante zaken te ontdekken, maar hun persoonlijke selectie kan natuurlijk maar een heel klein deel zijn van alle ontwikkelingen in de markt. Je kunt nog meer vinden op ons YouTube-kanaal “Elektor TV: Industry” op <https://youtube.com/@ElektorIndustry>.



Siglent

Op de stand van Siglent werd de DC Dual LISN van Elektor in gebruik genomen. De 5- μ H, 50- Ω Dual DC LISN ondersteunt spanningen tot 60 V en stromen tot 10 A. Hij meet RF-interferentie op beide kanalen door middel van 5- μ H smoorspoelen. Het interne 10dB-verzwakkings-netwerk (één voor elk kanaal) bevat een derde-orde hoogdoorlaatfilter met een kantelfrequentie van 9 kHz om de ingang van instrumenten zoals een spectrumanalyzer te beschermen tegen potentieel schadelijke gelijkspanningen of lage frequenties afkomstig van de te testen apparatuur.

Bekijk de demo op:
www.elektormagazine.com/ew-lisn



oxid[OS]

De programmeertaal C is al tientallen jaren de rigueur in de auto-industrie. Maar voor hoelang nog? We spraken met Flavia Oprea om uit te vinden waarom de taal Rust steeds meer in zwang komt en wordt gezien als een manier om de traditionele fouten te vermijden die inherent zijn aan embedded C-programmering. De startup, gevestigd in Roemenië, heeft een op Rust gebaseerd real-time besturingssysteem (RTOS) ontwikkeld dat op de beurs werd gedemonstreerd op een STM32-microcontroller. Automotive ontwikkelaars tonen nu al interesse dankzij het vooruitzicht van verbeterde beveiliging en de ondersteuning van geïsoleerde legacy C/C++ code.

<https://oxidos.io>

Bekijk het interview:

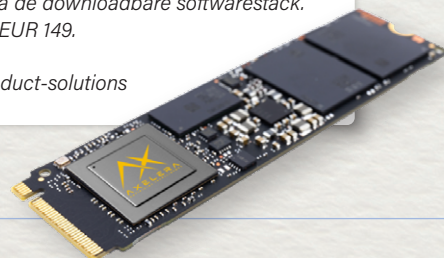
https://youtu.be/BNdvgJdN5_Q?t=1398



Axelera

Het Nederlandse bedrijf Axelera produceert modules die AI-inferentietoepassingen zoals beeldherkenning direct aan de edge kunnen verwerken. Boards zijn leverbaar in verschillende vormfactoren zoals M.2 (foto) of PCIe. Op basis van een enkele Metis AIPU, levert de M.2-module tot 214 tera-bewerkingen per seconde (TOP's) terwijl het stroomverbruik wordt geminimaliseerd en de integratie wordt vereenvoudigd via de downloadbare softwarestack. De prijs bedraagt USD/EUR 149.

<https://axelera.ai/ai-product-solutions>





Mikroelektronika

Op de beurs toonde Mikroe een met een camera uitgerust frame voor hun Planet Debug-systeem: ontwikkelaars en studenten kunnen samen op afstand een microcontroller-project ontwikkelen en debuggen, zonder dat ze dezelfde boards op hun bureau hoeven te hebben. Gebruikers kunnen nu hun hardware configureren conform hun eigen speciale behoeften door moederborden, processorkaarten en een periferie en displays te selecteren.

<https://mikroe.com/planet-debug>



Espressif

De ESP32 staat bekend om zijn gemakkelijke gebruik bij het implementeren van draadloze toepassingen. Daarom waren ESP32 en andere Espressif-oplossingen te zien in tientallen interessante demo's op embedded world 2023. Enkele voorbeelden: een ESP-ZeroCode-module voor slimme jaloezieën, ESP RainMaker-demo's en Matter WiFi-apparaten. Elektor sprak ook Amey Inamdar van Espressif bij embedded world om de laatste nieuwtjes te horen. Een van de hoogtepunten: RISC-V SoC's samen met een dual-core versie zit in de pijplijn; ondersteuning voor Matter om IoT-apparaten die verschillende protocollen gebruiken gemakkelijker te kunnen implementeren; en het RainMaker-platform voor het implementeren van de IoT cloud-backend.

Bekijk het vraaggesprek van
Stuart Cording!

<https://youtu.be/EFnUtAJX2aA>



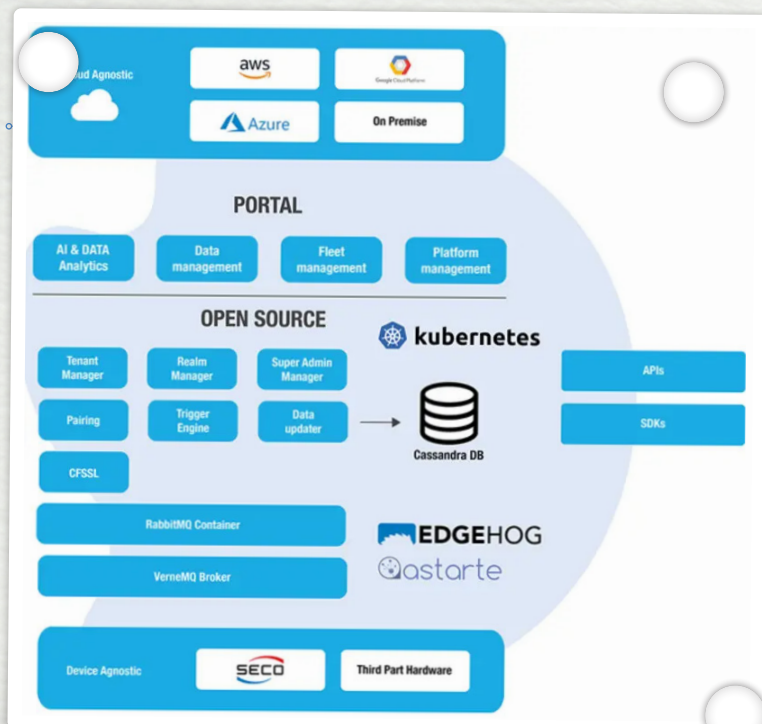
Live show mash-up

Voor de tweede keer kwamen Stuart Cording van Elektor Engineering Insights en Jens Nickel van Elektor Lab Talk bij elkaar om live verslag te doen van een elektronica-beurs. In de een uur durende show spraken Stuart en Jens met content director CJ Abate van Elektor over de interessantste zaken die te zien waren op de embedded world 2023 (zoals een cloud-gestuurde biertappende robot). Speciale gast: Elektor-auteur Viacheslav Gromov, die met zijn bedrijf AITAD de 'embedded award' in de categorie 'AI' won. ◀



Bekijk de volledige video hier:
www.youtube.com/watch?v=00d5TVKZx7Q





Seco

Het in Italië gevestigde IPC presenteert samen met Seco, leverancier van oplossingen, het modulaire IoT-platform CLEA. Aan de edge kan het event-data van de hardware zelf en van de applicaties van de klant bewaken. Voor dit doel biedt CLEA open API's en protocolconverters, evenals op AI gebaseerde edge-logica voor het analyseren van ruwe gegevens en afbeeldingen. CLEA is hardware- en cloud-onafhankelijk met een open-sourcebasis, waardoor beschikbaarheid op lange termijn wordt gegarandeerd. Reeds gerealiseerde toepassingen zijn onder meer slimme koffieautomaten, laadstations en medische hulpmiddelen.

<https://north.seco.com/en/products-and-services/clea-iot-platform>



Arm

Zoals we hebben besproken in onze Elektor Engineering Insights-show, verandert de wereld van de auto-industrie ingrijpend. Als de software voor een ECU eenmaal is voltooid, wordt deze tegenwoordig nooit meer bijgewerkt, tenzij er een veiligheidsprobleem wordt gevonden. Maar dat is aan het veranderen, zoals Robert Day uitlegde met het nieuwe SOAFEE-initiatief van Arm. Op basis van hoe orchestrators en containers worden gebruikt in cloud-software en CI/CD-ontwikkelingsprocessen, worden vergelijkbare methodologieën aangepast voor veiligheidskritische automotive software. Hier is momenteel een aantal leveranciers van halfgeleiders, auto-OEM's en hun leveranciers bij betrokken.

Bekijk het video-interview!
https://youtu.be/nr_6W4UgihM



Slint

Slint is een toolkit om op efficiënte wijze grafische gebruikersinterfaces te ontwikkelen voor embedded en desktop displays. Er worden meerdere programmeertalen ondersteund, waaronder Rust, C++ en JavaScript. Volgens de ontwikkelaars heeft de oplossing slechts een paar honderd kilobyte RAM nodig en weinig rekenkracht. Er zijn commerciële licenties, maar er is ook een open-source licentie onder GPLv3.

<https://slint-ui.com/#tryout>



De Digi XBee® XR 868-module is een compacte en betrouwbare oplossing die het gebruik van langereafstand-connectiviteitstoepassingen in de Europese regio ondersteunt. De vooraf gecertificeerde module werkt tussen 863 en 870 MHz en ondersteunt zowel point-to-point als mesh-netwerkprotocollen met een zichtbare bereik van meer dan 14 kilometer. Natuurlijk is er ook een dev-kit beschikbaar (foto).

<https://digi.com/products/embedded-systems/digi-xbee/rf-modules/sub-1-ghz-rf-modules/digi-xbee-xr-868>

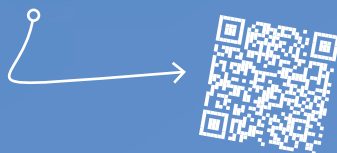
Elektor TV Shows



Elektor Engineering Insights

Elektor Industry Insights is een 'go-to' bron voor engineers en professionals die op de hoogte willen blijven over de wereld van de elektronica. Tijdens elke aflevering spreekt Elektor-redacteur Stuart Cording met experts uit de elektronica-industrie over technische uitdagingen en oplossingen.

www.elektormagazine.com/elektor-engineering-insights



Elektor LabTalk

Heb jij een passie voor DIY-elektronica, embedded systemen of algemene engineering theorie? Discussieer mee met het Elektor Labs team en onze redacteurs. Ze geven technische tips en bespreken toekomstige elektronica-projecten en de inhoud van Elektor Magazine. En uiteraard beantwoorden ze al jouw vragen!

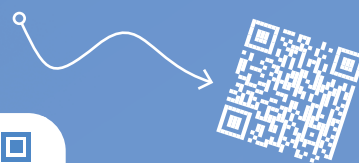
www.elektormagazine.com/elektor-lab-talk



elektor academy

Will jij jouw elektronikavaardigheden verbeteren? Dan is de Elektor Academy de 'place to be'. Onze expert Stuart Cording helpt u op weg via diverse cursussen over de meest uiteenlopende onderwerpen.

www.elektormagazine.com/elektor-academy



Blijf op de hoogte en volg
Elektor TV

www.youtube.com/c/ElektorIM



Sub-Nyquist-sampling in de praktijk

hogere frequenties betrouwbaar bemonsteren met subsampling

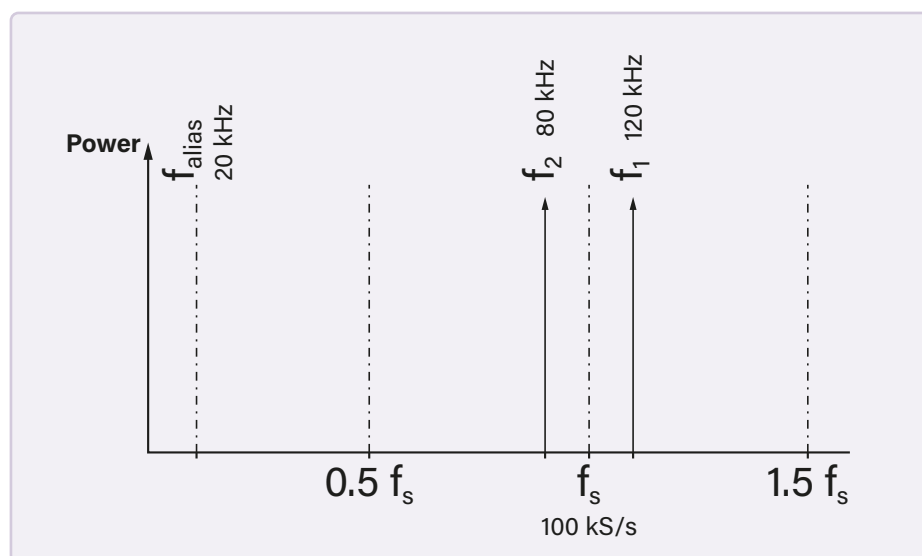
Sebastian Westerhold (AI5GW) (Duitsland)

Iedereen die te maken heeft met bemonsteringssystemen zal zich snel bewust worden van de beperkingen die worden opgelegd door het Nyquist-Shannon bemonsteringstheorema. Volgens deze regel kunnen signalen met frequenties hoger dan de helft van de bemonsteringsfrequentie niet meer betrouwbaar worden gedetecteerd. Dit is echter maar de halve waarheid. Een goede methode om signalen met een hogere frequentie te bemonsteren staat bekend als subsampling, ook wel sub-Nyquist-sampling genoemd. Dit artikel presenteert de methode op een praktische manier, met behulp van een eenvoudige Arduino Uno.

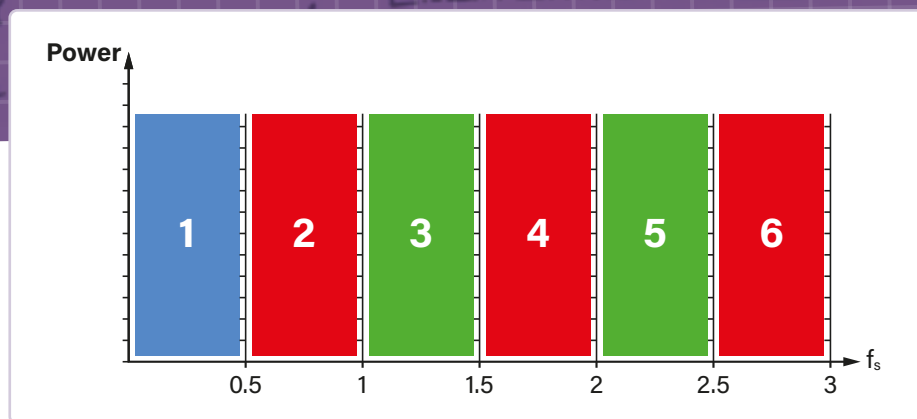
Het probleem

Het Nyquist-Shannon bemonsteringstheorema is uiterst belangrijk voor bemonsteringssystemen. De zeer vereenvoudigde versie ervan zegt dat de bemonsteringsfrequentie minstens tweemaal de hoogste frequentie in het te bemonsteren signaal moet bedragen. Zoiets heb je vast wel eens gehoord of gelezen.

Wat gebeurt er als je een zuiver sinusvormig signaal van 120 kHz bemonstert met een bemonsteringsfrequentie van slechts 100 kS/s? Dit resulteert in wat we aliasing bij 20 kHz noemen. De verkregen samples zien er dan uit alsof een sinusvormig signaal met een frequentie van 20 kHz (120 kHz – 100 kS/s) is bemonsterd. En aliasingfouten moeten met alle middelen worden vermeden. Of niet soms? Of aliasingfouten een storende factor zijn of daarentegen als nuttige techniek kunnen worden gebruikt, hangt af van de specifieke toepassing. In het voorbeeld met de bemonsteringsfrequentie van 100 kS/s zou het onmogelijk zijn een 20kHz-signaal te onderscheiden van een 120kHz-signaal; beide signalen zouden eruitzien als een 20kHz-signaal. Overigens zou een signaal van 80 kHz ook een aliasingsignaal van 20 kHz (100 kS/s – 80 kHz) opleveren (**figuur 1**). Deze omstandigheid is echter alleen van belang als een onderscheid tussen de genoemde signaalfrequenties echt nodig is. Als alle daadwerkelijk in het signaal voorkomende frequentiecomponenten binnen een duidelijk herkenbaar bereik vallen, kan zinvol worden bemonsterd terwijl aliasingeffecten opzettelijk worden uitgebuit. De eenduidigheid is gegeven als alle in het te bemonsteren signaal voorkomende frequentiecomponenten binnen dezelfde Nyquist-zone vallen. Het frequentiebereik van 0 Hz (DC) tot de helft van de bemonsteringsfrequentie (f_s) wordt Nyquist-zone 1 genoemd (**figuur 2**). Deze zone is het 'normale' werkgebied voor



Figuur 1. Bij een bemonsteringsfrequentie van 100 kS/s veroorzaken zowel een signaal van 120 kHz (f_1) als een signaal van 80 kHz (f_2) een aliasingeffect (f_{ALIAS}) van 20 kHz.



Figuur 2. Nyquist-zones 1...6. Zone 1 is het 'normale' bereik, zones 2, 4 en 6 zijn inverterende bereiken, zones 3 en 5 = niet-inverterende zones.

een bemonsteringssysteem. Nyquist-zone 2 strekt zich uit van $0,5 f_s$ tot f_s . Deze zones gaan dan theoretisch oneindig door met intervallen van $0,5 f_s$. De Nyquist-zones met even nummers hebben een bijzondere eigenschap: binnen deze zones vindt een inversie van het frequentiespectrum plaats.

Als de bandbreedte van een te detecteren signaal dus beperkt is tot het bereik van een van deze Nyquist-zones, kan subsampling worden gebruikt om signalen te detecteren waarvan de frequentiecomponenten de bemonsteringsfrequentie aanzienlijk overschrijden.

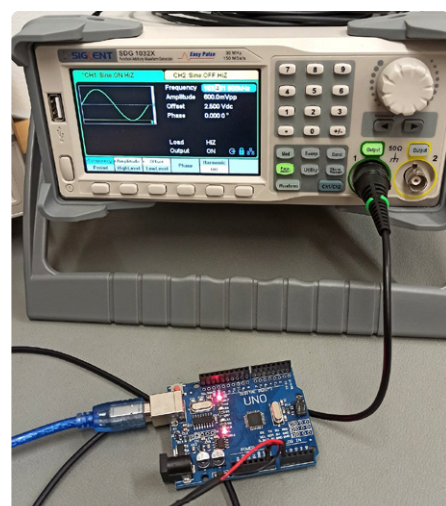
Een praktijkvoorbeeld

Voor een project moet de frequentie van een signaal in een bereik van ongeveer 160 kHz tot 165 kHz betrouwbaar worden gedetecteerd met een Arduino Uno [2] (figuur 3). Om technische redenen is de bemonsteringsfrequentie ingesteld op ongeveer 154 kS/s met behulp van de betreffende registers (prescaler 1:8) (strikt genomen is dat 153,846 S/s; voor de duidelijkheid wordt hieronder de afgeronde waarde gebruikt). De signaalfrequentie is dus enkele kHz hoger dan de bemonsteringsfrequentie zelf, maar kleiner dan $1,5 f_s$ (overeenkomend met

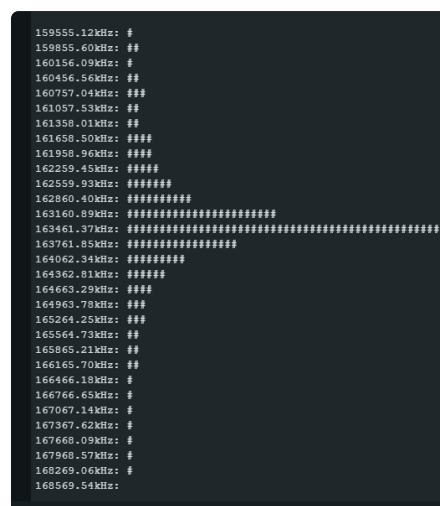
Nyquist-zone 3). Daardoor ligt de te verwachten aliasing als gevolg van subsampling tussen 6 kHz ($160 \text{ kHz} - f_s$) en 11 kHz ($165 \text{ kHz} - f_s$). De Arduino-code is zo geschreven dat de A/D-converter van de Arduino een interruptroutine (ISR) gebruikt om het signaal op analoge ingang 0 te bemonsteren totdat een buffer met 512 discrete samples is gevuld. De (aliasing-)signaalfrequentie wordt dan uit deze waarden berekend voordat de volgende 512 samples worden ingelezen en het spel opnieuw begint.

Om de frequentie te bepalen is het Goertzel-algoritme gebruikt, een speciale vorm van de discrete Fouriertransformatie die zuinig met resources omgaat [1]. Het resultaat wordt vervolgens voor demonstratiedoeleinden uitgevoerd via de seriële interface en visueel geformatteerd voor de seriële monitor (figuur 4).

Volgens hetzelfde principe zou je bijvoorbeeld een IF van 455 kHz van een radio-ontvanger in Nyquist-zone 6 ($3 f_s$, overeenkomend met ongeveer 462 kHz) met subsampling kunnen converteren en FSK-signalen (NAVTEXT, RTTY enzovoort) zelfs met de niet extreem krachtige Arduino Uno kunnen demoduleren.



Figuur 3. Testopstelling met Arduino en signaalgenerator.



Figuur 4. De resultaten worden naar de seriële monitor gestuurd.

Opmerkingen

Om subsampling naar wens te laten werken, moet de gebruikte ADC de vereiste analoge bandbreedte hebben. Deze is gewoonlijk veel groter dan de bemonsteringsfrequentie. In de datasheet vind je deze waarde meestal als "Full power bandwidth". Met de ATmega328P lijkt subsampling heel netjes te werken tot in het MHz-bereik, zoals mijn eigen experimenten suggereren. Helaas leverden mijn experimenten met de Raspberry Pi Pico geen veelbelovende resultaten op. ◀

vertaling: Willem den Hollander — 220629-03

Vragen of opmerkingen?

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



Gerelateerde producten

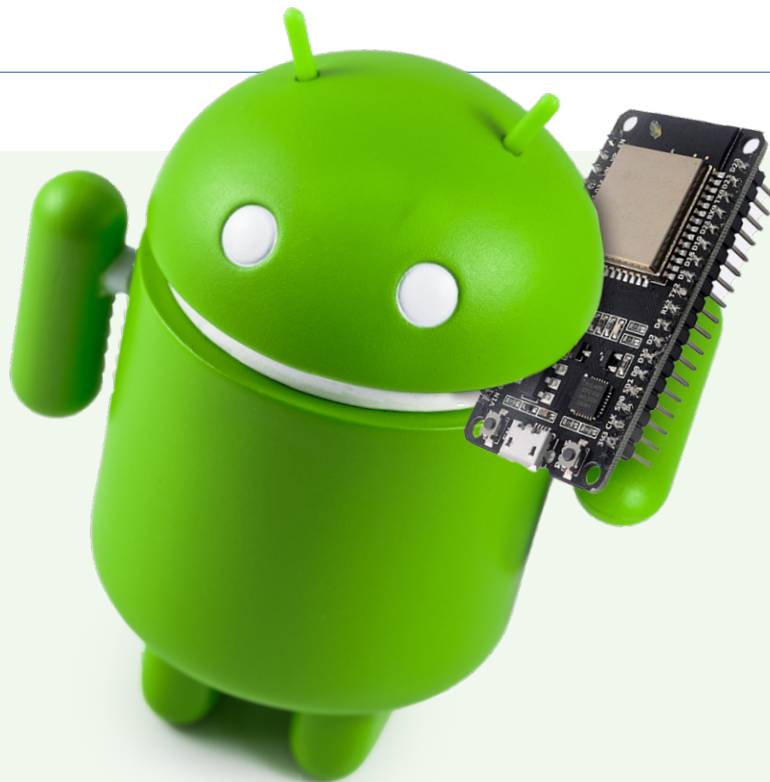
- > **Siglent SDG1032X 2-ch Signal Generator (30 MHz) (SKU 20276)**
www.elektor.nl/20276
- > **OWON XSA810 Spectrum Analyzer (1 GHz) (SKU 19714)**
www.elektor.nl/19714
- > **HackRF One Software Defined Radio (1 MHz to 6 GHz) (SKU 18306)**
www.elektor.nl/18306
- > **MonoDAQ-U-X Multifunctional USB Data Acquisition System (50 kS/s) (SKU 18766)**
www.elektor.nl/18766

WEBLINKS

- [1] Sebastian Westerhold (2022) – Goertzel-bibliotheek voor Arduino:
<https://github.com/AI5GW/Goertzel>
- [2] Arduino-sketch bij dit artikel:
<http://www.elektormagazine.nl/220629-03>

Met Android-smartphone, spreek ik met ESP32?

praktisch project met behulp van de Android WiFi-API



Tam Hanna (Hongarije)

Ontwikkelaars van Android-apps weten dat de code die nodig is om verbinding te maken met een WiFi-netwerk niet triviaal is, vooral wanneer je het hele proces voor de eindgebruiker zo soepel en vlot mogelijk wilt laten verlopen. Software-ontwikkelaars moeten er bovendien voor zorgen dat hun code compatibel is met verschillende Android-versies en op de hoogte zijn van de door Google vereiste beveiligingsmaatregelen, zodat de app probleemloos functioneert. Dit artikel wil je helpen een weg te vinden door deze jungle.

Als je je geen zorgen maakt over het relatief hoge energieverbruik, dan is WiFi de radiostandaard bij uitstek, vooral voor kleinere projecten. Bluetooth komt niet in aanmerking vanwege de certificerings- en licentiekosten van Bluetooth SIG. Communicatie kan plaatsvinden via een REST interface-architectuur, en WiFi-zenders zijn zowat overal verkrijgbaar.

Het is geen verrassing dat kleine ESP32-ontwikkelboards – met hun ingebouwde WiFi access points – zo populair zijn bij ontwikkelaars van Android-toepassingen.

Het Android OS is zo ontworpen dat de ervaring van de gebruikers-interface voorrang krijgt op de reactietijd op externe stimuli. Deze hoge latentie maakt het ongeschikt voor het besturen van tijdkritische gebeurtenissen. Android is uitgerust met uitgebreide GUI-stacks, en voor veeleisende taken zoals het weergeven van grafieken zijn er bibliotheken die via Gradle geladen kunnen worden.

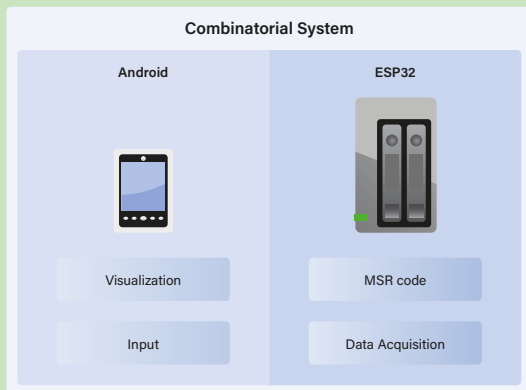
In het geval dat een systeem een snelle reactie op real-time gebeurtenissen nodig heeft en ook UI-vereisten heeft, lijkt het logisch om de verwerking te verdelen zoals geschetst in **figuur 1**.

Vaak is de zwakke schakel in de keten de eindgebruiker, die uiteindelijk waarden in het systeem moet invoeren. Als de eindgebruiker 'handmatig' een verbinding met een bepaald WiFi-netwerk tot stand moet brengen, wordt uw product-ondersteuningsteam gegarandeerd overspoeld met telefoontjes van gefrustreerde klanten.

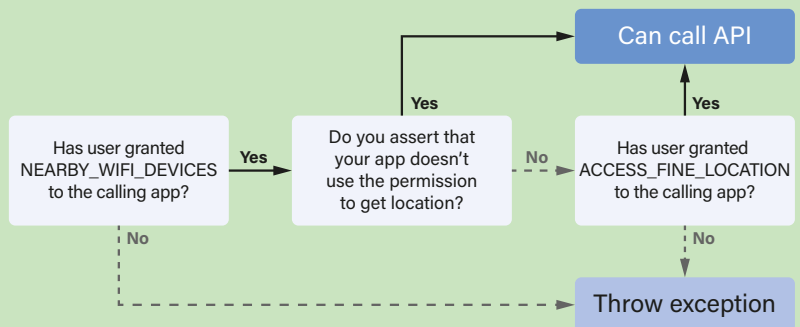
Dat zou vlotter kunnen als een automatische aansluiting op een netwerk tot stand zou kunnen worden gebracht. Google maakt zich echter zorgen over de functionele veiligheid en de bescherming van persoonsgegevens bij een dergelijke techniek.

Wat willen we?

Het doel van het volgende artikel is het beschrijven van de Android-code die nodig is om een communicatieverbinding tot stand te brengen met een WiFi-doelnetwerk. Een ESP32 draait SoftAP om een remote WiFi access point op te zetten, waarmee het Android-apparaat verbinding kan maken. In de praktijk kan deze code ook worden gebruikt met veel andere controllers met WiFi-capaciteit (zie de download op [1]). Dit artikel is gebaseerd op een project dat de auteur ontwikkelde voor een klant. Vrijwel elk ESP32-board dat een variant van het voorbeeldprogramma `esp-idf/examples/wifi/getting_started/` (opgenomen in de ESP-IDF [2]) draait, geconfigureerd als toegangspunt, zou ook geschikt zijn. We veronderstellen dat de lezer ervaring heeft met de ESP32-ontwikkelomgeving.



Figuur 1. De Android-smartphone verzorgt de GUI, terwijl de ESP32 verantwoordelijk is voor de hardware-communicatie.



Figuur 2. Voor draadloze communicatie onder Android 13 kunnen extra machtigingen nodig zijn (zie afbeelding bij [4]).

Aan de Android-kant moet je twee smartphones hebben: één met een Android-versie ≥ 10 , en één met Android 9 of eerder. Android Studio [3] dient als ontwikkelomgeving. De ruimte is hier beperkt, dus ik ga ervan uit dat de lezer enige ervaring heeft met de Android IDE.

Configuratie

De compilatie-configuratie van een Android-project is een vrij ingewikkeld proces. Meestal is er een versie van het *build.gradle*-bestand dat bij de *app*-module hoort, zodat alle noodzakelijke instellingen en afhankelijkheden van de app worden ingesteld in Android Studio. Dit bestand bevat typisch statements zoals:

```

android {
    compileSdkVersion 29
    buildToolsVersion "30.0.1"
    defaultConfig {
        minSdkVersion 26
        targetSdkVersion 31
    }
}

```

Met het veld `minSdkVersion` kan de ontwikkelaar de minimale versie van het Android-besturingssysteem opgeven voor het doel- of eindapparaat waarmee de app compatibel is. De waarde 26 verwijst naar Android 8 (Oreo). `compileSdkVersion` specificeert de versie van de Software Development Kit waarmee de app zal worden gecompileerd, en `buildToolsVersion` geeft aan welke versie van de Android build tools zal worden gebruikt om het project te compileren. De regel `targetSdkVersion` specificeert de versie van het Android OS waarvoor een app wordt ontwikkeld.

Dit onderscheid, dat op het eerste gezicht ingewikkeld en verwarrend klinkt, is nodig omdat het 'gedrag' van veel Android API's verandert afhankelijk van de OS-versie waaronder het draait.

Als `targetSdkVersion` is ingesteld op een bepaalde waarde, gaat het besturingssysteem ervan uit dat de ontwikkelaar bij het ontwerp van de app rekening heeft gehouden met alle wijzigingen die in deze versie zijn geïntroduceerd – als de waarde kleiner is, wordt de compatibiliteitsmodus geactiveerd.

Helaas heeft Google strikte richtlijnen opgesteld voor de waarden in `targetSdkVersion`. Als een app is ontworpen om alleen op oudere OS-versies te draaien, zal Android Studio je tijdens het compileren waarschuwen met een bericht in de vorm van 'Google Play requi-

res that apps target API level 30 or higher', wat aangeeft dat de Play Store-backend zal weigeren de gegenereerde APK te uploaden. In de niet al te verre toekomst wil Google 'zeer oude' apps geleidelijk uit de Play Store verwijderen.

Als ontwikkelaar moet je je bewust zijn van deze beperkingen en mogelijke valkuilen.

Machtigingen

Je kunt de wereld van Android ruwweg verdelen in de tijd voor en de tijd na het nieuwe permission management system en/of Storage Access Framework: in de goede oude tijd waren ontwikkelaars over het algemeen vrij om te doen wat ze wilden, maar toen ging Google serieus werk maken van de beveiliging van persoonsgegevens.

In het geval van WiFi is de situatie netelig, aangezien informatie over de standplaats kan worden afgeleid uit de kennis van de draadloze netwerken in de omgeving (de eerste iPod touch van Apple, zonder GPS of mobiele telefoon, kon bijvoorbeeld op deze manier zijn locatie bepalen). Het is nu nodig om de volgende declaraties in het manifest-bestand van de Android-app te specificeren om te verzoeken dat de app bepaalde machtigingen krijgt.

```

<uses-permission android:name=
"android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name=
"android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name=
"android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name=
"android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name=
"android.permission.CHANGE_NETWORK_STATE" />
<uses-permission android:name=
"android.permission.ACCESS_NETWORK_STATE" />

```

Als je Android 13 in je toepassing wilt ondersteunen, moet je een extra wijziging aanbrengen: in het geval van WLAN werkt de Android 13 machtigingslogica zoals geschetst in **figuur 2**.

Het is nu nodig om de volgende declaraties in het manifest-bestand van de Android-app te specificeren om te verzoeken dat de app bepaalde machtigingen krijgt.

```
<uses-permission android:name=
"android.permission.NEARBY_WIFI_DEVICES"
android:usesPermissionFlags="neverForLocation" />
```

De `android.permission.NEARBY_WIFI_DEVICES` machtiging is een verzoek van de app-ontwikkelaar om de app toe te staan te scannen naar en toegang te krijgen tot alle WiFi-apparaten in de buurt, terwijl `android:usesPermissionFlags="neverForLocation"` een speciaal regime is dat een 'nooit voor locatie'-vlag toekent aan de app. Dit verzekert dat de app WiFi-informatie alleen gebruikt voor configuratiedoeleinden en niet voor locatie-informatie. In dit geval is de machtiging `android.permission.ACCESS_FINE_LOCATION` niet vereist onder Android 13. Eerdere versies van het besturingssysteem vereisen deze machtiging, in welk geval een configuratie zoals deze wordt aanbevolen:

```
<uses-permission android:name=
"android.permission.ACCESS_FINE_LOCATION"
android:maxSdkVersion="32" />
```

In deze context garandeert het attribuut `maxSdkVersion` dat de toestemmingsdeclaratie alleen door eerdere Android-versies wordt herkend.

Tweesporen-ontwikkeling

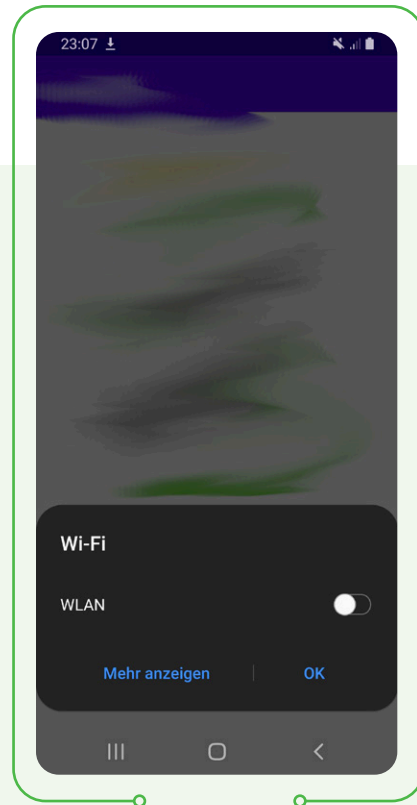
Bij het instellen van de *Activity* die verantwoordelijk is voor het 'scannen' van de omgeving naar een bruikbaar WiFi-eindpunt, moeten we eerst controleren of de WiFi van de telefoon is ingeschakeld:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    ...
    if (android.os.Build.VERSION.SDK_INT <
        Build.VERSION_CODES.Q) {
        wifiManager =
            (WifiManager) getApplicationContext().
                getSystemService(Context.WIFI_SERVICE);
        if (!wifiManager.isWifiEnabled()) {
            wifiManager.setWifiEnabled(true);
        }
    }
}
```

Als we te maken hebben met een oudere versie van Android die niet wordt beïnvloed door API-wijzigingen, kan ons programma de WiFi van de telefoon inschakelen zonder om toestemming van de gebruiker te vragen. Helaas geldt dit niet voor nieuwere versies van Android – bij deze versies is toestemming van de gebruiker vereist.

Om in runtime onderscheid te maken tussen de twee varianten, vertrouwen we op de constante `android.os.Build.VERSION.SDK_INT`. Deze verwijst naar de API-versie die wordt ondersteund door de firmware van de huidige telefoon.

Om WiFi daadwerkelijk in te schakelen, moeten we een *intent* sturen van het type `Settings.Panel.ACTION_WIFI`:



Figuur 3. Google vraagt uitdrukkelijke toestemming van de gebruiker.

```
else {
    wifiManager =
        (WifiManager) getApplicationContext().
            getSystemService(Context.WIFI_SERVICE);
    if (!wifiManager.isWifiEnabled()) {
        Toast toast =
            Toast.makeText(getApplicationContext(),
                "PLEASE enable WiFi so that we can connect!",
                Toast.LENGTH_LONG);
        toast.show();
        Intent panelIntent =
            new Intent(Settings.Panel.ACTION_WIFI);
        this.startActivityForResult(panelIntent,
            ICY_WIFI_REQCODE);
    }
}
```

Je wordt voor alle moeite beloond met het bericht in **figuur 3** dat in runtime verschijnt. De gebruiker moet de knop omzetten om het activeringsproces te autoriseren.

Het hier afgedrukte fragment vuurt een *toast* af naar de gebruiker voordat de *Intent* wordt verzonden, met de mededeling dat WiFi moet worden geactiveerd.

Indien geen actie wordt ondernomen, kan het nodig zijn de gebruiker eraan te herinneren dat hij WiFi moet activeren.

Voor Android-beginners kan het vreemd lijken dat `ICY_WIFI_REQCODE` wordt doorgegeven aan `startActivityForResult()`, maar dit is een correlatiecode waarmee het ontvangende eindpunt de trigger kan identificeren die verantwoordelijk is voor het inkomende verzoek. Correlatiecodes zijn normale integers waaraan het besturingssysteem

Goud van oud

Als je klaar bent met je eerste smartphone-app, moet je eens even teruggaan in de tijd en de basisprincipes bekijken die Jeff Hawkins gebruikte om Palm OS te maken, dat draaide op Palm- en Palm Pilot-toestellen. Ook al zijn ze nu verouderd, zijn filosofie bij het ontwerpen van Palm OS is interessant. *Zen of Palm* is niet erg lang, en je kunt een PDF-versie vinden op [5].



geen significante waarde toekent. De auteur declareert ze in het programma volgens het volgende schema; het enige dat telt is dat de gegenereerde code uniek is.

```
public class MainActivity extends
    AppCompatActivity implements
    AdapterView.OnItemClickListener {
    private final int MY_PERMISSIONS_
        ACCESS_COARSE_LOCATION = 1;
    int ICY_WIFI_REQCODE = 4242;
```

De volgende taak is het eigenlijke WiFi-scanproces te starten. Door ruimtegebrek kunnen we niet in de Android GUI-stack duiken, maar we vertrouwen erop dat je een manier zult vinden om de `getWifi()`-methode te activeren.

De eerste stap in de verwerking van standplaatsinformatie is controleren of we al access-machtiging hebben tot de `Manifest.permission.ACCESS_FINE_LOCATION`. Onder Android 13 moet je ook de nieuwe machtiging opnemen in de analysecode:

```
private void getWifi() {
    if (ContextCompat.
        checkSelfPermission(MainActivity.this,
            Manifest.permission.ACCESS_FINE_LOCATION) !=
            PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.
            requestPermissions(MainActivity.this,
                new String[],
                MY_PERMISSIONS_ACCESS_COARSE_LOCATION);
    }
```

Vergeet niet dat, voor Android 6.0 en hoger, 'gevoelige' machtigingen alleen door het besturingssysteem worden ingeschakeld als de gebruiker daar uitdrukkelijk mee instemt. Hiervoor worden dialogen gebruikt zoals die in **figuur 4**.

Als we de toestemming al hebben gegeven, is de volgende stap nog een versiewissel. Als we te maken hebben met een eerdere versie van Android, roepen we direct de methode `getWifiWorkerOld()` aan:

```
else {
    if (android.os.Build.VERSION.SDK_INT <
        Build.VERSION_CODES.Q) {
        getWifiWorkerOld();
    }
```

In het geval van een nieuwe telefoon wordt, om een crash te voorkomen, op dit punt de voeding van de WiFi-zender opnieuw gecontroleerd, omdat het denkbaar is dat gebruikers WiFi uitschakelen voordat de methode actief wordt.

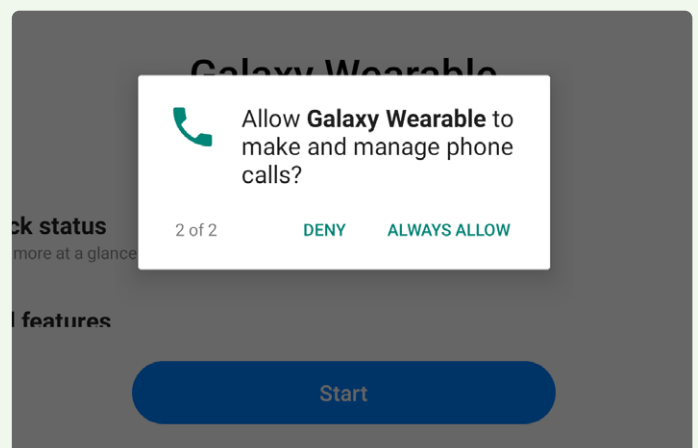
```
else {
    if (wifiManager.isWifiEnabled()) {
        getWifiWorkerOld();
    }
    else {
        Toast toast =
            Toast.makeText(getApplicationContext(),
                "PLEASE enable WiFi so that we can connect!",
                Toast.LENGTH_LONG);
        toast.show();
        Intent panelIntent =
            new Intent(Settings.Panel.ACTION_WIFI);
        this.startActivityForResult(panelIntent,
            ICY_WIFI_REQCODE);
    }
}
```

Zodra we met succes de voedingstoestand van de zender hebben gecontroleerd, gaan we over tot het activeren van het scanproces met behulp van deze code:

```
private void getWifiWorkerOld() {
    ...
    wifiManager.startScan();
}
```

`wifiManager` is een systeemklasse die we verkregen bij het activeren van de activiteit. De methode `startScan()` zorgt vervolgens voor het starten van het scanproces.

Het terugzenden van de informatie die door de scanrun wordt opgevangen, gebeurt via *Broadcast*. Voor de ontvangst ervan hebben we een *BroadcastReceiver* nodig. Omdat we die niet willen aanmaken in het manifest-bestand, schrijven we hem in plaats daarvan in de methode `onPostResume()` van de activiteit:



Figuur 4. Let op het nieuwe machtigingensysteem.


```

@Override
protected void onResume() {
    super.onResume();
    receiverWifi =
        new WifiReceiver(wifiManager, wifiList, this);
    IntentFilter intentFilter =
        new IntentFilter();
    intentFilter.addAction
        (WifiManager.SCAN_RESULTS_AVAILABLE_ACTION);
    registerReceiver(receiverWifi, intentFilter);
}

```

In het belang van de compliantie met de Play Store is het belangrijk dat handmatig geregistreerde *BroadcastReceivers* ook door het besturings-systeem worden afgemeld. Dit kan het gemakkelijkst worden gedaan met behulp van een van de levenscyclusmethoden van de activiteit. Ik heb ervoor gekozen hier de methode `onPause()` te gebruiken:

```

@Override
protected void onPause() {
    super.onPause();
    unregisterReceiver(receiverWifi);
}

```

Broadcast Receiver voor gegevensverwerking

Aangezien de ontvangst van netwerk informatie die door het WiFi scanproces wordt geretourneerd, gebeurt via het broadcast-systeem van Android, moeten we de klasse *BroadcastReceiver* implementeren die hierboven bij het besturingssysteem is geregistreerd.

BroadcastReceivers in Android hebben de vorm van klassen die zijn afgeleid van de *BroadcastReceiver* hoofdklasse. In ons voorbeeld – de GUI-variabelen negerend – ziet de 'header' er als volgt uit (we drukken de constructor die nodig is om de velden te vullen hier niet af):

```

class WifiReceiver extends BroadcastReceiver {
    WifiManager wifiManager;
    ListView wifiDeviceList;
    MainActivity myParentAct;
}

```

De belangrijke vraag is nu hoe de netwerk informatie verwerkt moet worden die terugkomt van de WiFi-zenders. Het antwoord is te vinden in de methode `onReceive()`, die het besturingssysteem activeert wanneer de uitgezonden informatie, die wordt geleverd in de vorm van Intents, arriveert.

```

public void onReceive(Context context, Intent intent) {
    String action = intent.getAction();
    if (WifiManager.
        SCAN_RESULTS_AVAILABLE_ACTION.equals(action)) {
        List<ScanResult> wifiList =
            wifiManager.getScanResults();
        ArrayList<String> deviceList =
            new ArrayList<>();
        for (ScanResult scanResult : wifiList) {

```

```

            if(scanResult.SSID.
                contains("NAMEOFTHING"))
                deviceList.add(scanResult.SSID );
        }
        myParentAct.myArrayAdapter=
            new ArrayAdapter(context,
                android.R.layout.simple_list_item_1,
                deviceList.toArray());
        wifiDeviceList.
            setAdapter(myParentAct.myArrayAdapter);
    }
}

```

In de eerste stap roept de code `intent.getAction()` aan om de naam van de Intent te controleren. *Broadcast Receiver* kan theoretisch elke intent bevatten; door te controleren aan de hand van de string `WifiManager.SCAN_RESULTS_AVAILABLE_ACTION`, zorgen we ervoor dat we te maken hebben met een 'compatibele' intent.

Aangezien het codevoorbeeld waaraan de auteur de volgende fragmenten ontleende een lijst gebruikt voor de weergave, is in de volgende stap enige aanpassing vereist. Hoe dan ook, de zin van het proces is de populatie van de klasse `deviceList`.

In een echte toepassing zou op dit punt een lijst worden gevuld met informatie, maar je kunt natuurlijk op dit punt anders te werk gaan als je slechts een enkel WiFi-doelnetwerk als geldig beschouwt.

Een voorbeeld van de andere aanpak is een apparaat dat tijdens de configuratie contact maakt met zijn basistoepassing en dit aangeeft met een specifieke WiFi-naam. In dat geval zouden we direct na het detecteren van dat netwerk verder kunnen gaan.

Een nette verbindingsssetup

Gebruikers houden niet van lange wachttijden bij smartphone-toepassingen. Daarom is het zinvol voortgangsbalken of soortgelijke gadgets te tonen die de gebruiker verzekeren dat de toepassing niet is vastgelopen.

In de client-toepassing die als basis dient voor dit artikel, is dit geïmplementeerd in de vorm van een speciale activity: deze toont een bedrijfslogo en een voortgangsbalk, en zorgt zo voor enige geruststelling. Hij wordt geactiveerd door een intent te versturen:

```

public void onItemClick(AdapterView<?>
    adapterView, View view, int pos, long anID) {
    String y = (String) myArrayAdapter.getItem(pos);
    Intent intent =
        new Intent(MainActivity.this,ConnActivity.class);
    intent.putExtra("WIFINAME", y);
    startActivity(intent);
}

```

In 'Code Behind' begint het met het declareren van een *AsyncTask* die een aanwezigheidstest uitvoert. Dit is een verificatie (hier niet beschreven) dat de connection peer deel uitmaakt van het client-ecosysteem:

Pas op voor de Google-blokkade

Vanaf Android 11 heeft Google 'eenmalige machtigingen' ingevoerd, waarmee de gebruiker een app eenmalig toegang kan verlenen tot een bepaalde functie of gegevens. Dit verandert ook de manier waarop apps op de achtergrond toestemming vragen, waardoor sommige verzoeken kunnen worden geblokkeerd. De redenering hierachter is om gebruikers meer controle over hun gegevens te geven en de privacy op Android te verbeteren. Ontwikkelaars moeten zich bewust zijn van deze potentiële problemen.



```
public class ConnActivity extends AppCompatActivity {
    PresenceTestAsynctask aT;
    ConnActivity myself;
```

In `onCreate()` begint de verbindingsopbouw zonder verdere actie van de gebruiker. De eerste officiële actie is het configureren van enkele member-variabelen en het aanroepen van de `getIntent().getExtras()` methodes om de informatie – eerder 'verpakt' door Intent – toegankelijk te maken met de WiFi-naam en andere ondersteunende gegevens:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_conn);
    myself=this;
    Bundle b = getIntent().getExtras();
    String networkPass = "sdsds";
```

De volgende stap is nog een versiewissel die de 'versiestatus' van de doel-hardware controleert.

Als we te maken hebben met een eerdere versie van Android, is de volgende stap de start van de verbindingsopbouw volgens de onderstaande code:

```
if (android.os.Build.VERSION.SDK_INT <
    Build.VERSION_CODES.Q) {
    WifiConfiguration conf =
        new WifiConfiguration();
    conf.SSID = "\"" +
        b.getString("WIFINAME") + "\"";
    conf.preSharedKey = "\"" + networkPass + "\"";
    WifiManager wifiManager =
        (WifiManager) getApplicationContext().
        getSystemService(Context.WIFI_SERVICE);
    wifiManager.addNetwork(conf);
```

De methode `addNetwork()` neemt een object van het type `WifiConfiguration` over. Diens taak is het leveren van een complete WLAN-informatieset, waarmee de telefoon vervolgens verbinding kan maken.

In onze laatste handeling moeten we de verbinding opzetten volgens het volgende schema, wat gebeurt met drie methoden: `disconnect()`, `enableNetwork()` en `reconnect()`. Daarna volgt de activering van de AsyncTask – als het externe station met succes is geauthenticeerd, begint de clienttoepassing met de eigenlijke configuratie van de aangesloten hardware:

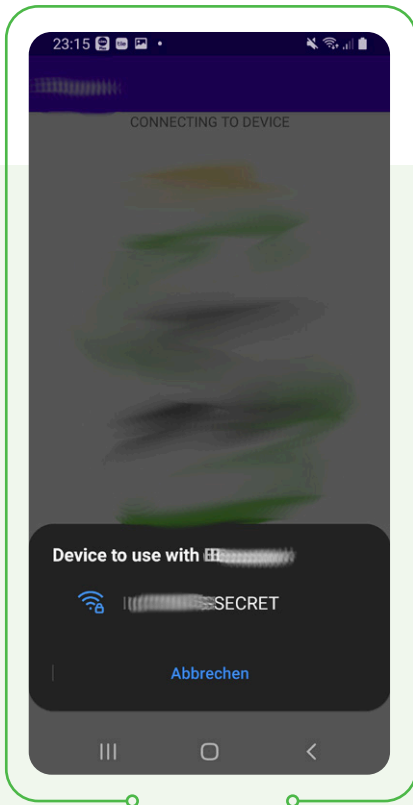
```
//Can only come here from permission granted state
@SuppressLint("MissingPermission")
List<WifiConfiguration> list =
    wifiManager.getConfiguredNetworks();
for (WifiConfiguration i : list) {
    if (i.SSID != null && i.SSID.equals("\"" +
        b.getString("WIFINAME") + "\"")) {
        final ConnectivityManager connectivityManager =
            (ConnectivityManager) getApplicationContext().
            getSystemService(Context.CONNECTIVITY_SERVICE);
        ConnectivityManager.NetworkCallback
            networkCallback =
            new ConnectivityManager.NetworkCallback() {
                @Override
                public void onAvailable(@NonNull Network network) {
                    super.onAvailable(network);
                    //ONLY IF ANDROID 9
                    if (android.os.Build.VERSION.SDK_INT ==
                        Build.VERSION_CODES.Q) {
                        connectivityManager.
                            bindProcessToNetwork(network);
                    }
                }
            };
        connectivityManager.
            registerDefaultNetworkCallback(networkCallback);

        wifiManager.disconnect();
        wifiManager.enableNetwork(i.networkId, true);
        wifiManager.reconnect();
        aT = new PresenceTestAsynctask
            (getApplicationContext());
        aT.myAct=mySelf;
        aT.execute("...");
        break;
    }
}
```

Bij nieuwere versies van Android is op dit punt opnieuw interventie van de gebruiker vereist. Het werk begint met het aanmaken van een `NetworkSpecifier`-object zoals dit:

```
}else{
    final NetworkSpecifier specifier =
        new WifiNetworkSpecifier.Builder()
            .setSsidPattern(new PatternMatcher(
                b.getString("WIFINAME"),
                PatternMatcher.PATTERN_PREFIX))
            .setWpa2Passphrase(networkPass)
            .build();
```

`NetworkSpecifier`-objecten zijn een soort zoekfilter die de Android GUI helpt draadloze netwerk-peers te identificeren die relevant zijn voor het specifieke gebruiksgaueval.



Figuur 5. Android Q vraagt de gebruiker alvorens een verbinding tot stand te brengen.

```
connectivityManager.  
    bindProcessToNetwork(network);  
aT = new PresenceTestAsyncTask  
    (getApplicationContext());  
aT.myAct=mySelf;  
aT.execute(". . .");  
}  
@Override  
public void onUnavailable() {  
    super.onUnavailable();  
}  
};
```

Als de methode `onAvailable()` wordt aangeroepen, is de verbinding met succes tot stand gebracht. In dat geval start de toepassing de `AsyncTask` opnieuw om de remote end te authenticeren en indien nodig aanvullende acties uit te voeren.

De laatste actie van onze scanmethode is dan het aanroepen van `requestNetwork()`, dat de hierboven getoonde gebruikersinterface activeert.

```
connectivityManager.  
    requestNetwork(request, networkCallback);  
}  
}  
}
```

De volgende stap is het bouwen van een `NetworkRequest`-element, hetgeen als volgt gebeurt:

```
final NetworkRequest request =  
    new NetworkRequest.Builder()  
        .addTransportType(NetworkCapabilities.  
            TRANSPORT_WIFI)  
        .removeCapability(NetworkCapabilities.  
            NET_CAPABILITY_INTERNET)  
        .setNetworkSpecifier(specifier)  
        .build();
```

De eigenlijke setup van de verbinding vindt plaats zoals getoond in **figuur 5**. De gebruiker moet de naam invoeren van het WiFi-netwerk dat het configuratiedoel wordt.

Een neveneffect van deze procedure is dat de WiFi-verbinding asynchroon tot stand moet worden gebracht. Feedback over succes en mislukking wordt gegeven via een callback die moet worden aangeemaakt in de toepassing die verantwoordelijk is voor de activering:

```
final ConnectivityManager connectivityManager =  
(ConnectivityManager)getApplicationContext().  
getSystemService(Context.CONNECTIVITY_SERVICE);  
final ConnectivityManager.NetworkCallback  
    networkCallback =  
    new ConnectivityManager.NetworkCallback() {  
        @Override  
        public void onAvailable(@NonNull Network network) {  
            super.onAvailable(network);
```

Is het al die moeite waard?

Er zijn vaak veel manieren om een engineering-probleem op te lossen. Het is waar dat real-time besturingssystemen – ik denk hier aan het Azure RTOS – samen met hun ingebodde GUI-stacks en draaiend op een microcontroller, inderdaad krachtige grafische interfaces kunnen realiseren in een single-device oplossing die de meeste geavanceerde gebruikssituaties zal dekken.

Om dezelfde functionaliteit te bereiken als met de hier beschreven Android-configuratie, zouden hogere hardwarekosten ontstaan – er komt geen touchscreen voor de GUI, aangezien de bijbehorende logica, zoals een framebuffergeheugen, teveel onderdelen zou vergen. Bovendien moeten we denken aan de tijd en moeite die nodig zijn om het systeem te ontwikkelen: als je bijvoorbeeld de overdracht van beelden en sensormetingen per e-mail zou willen implementeren in een real-time besturingssysteem, moet je een paar dagen uittrekken voor het coderen en je ervan bewust zijn dat de extra belasting gevolgen kan hebben voor de systeematentie. De Arduino-combinatie biedt een gelikte GUI op een apparaat dat de meesten van ons al bij zich dragen. Het idee om je telefoon tevoorschijn te halen en aan te sluiten op een lokaal toegangspunt, om wat voor reden dan ook, is best wel cool.

Als je dit project wilt gaan ontwikkelen, hoop ik dat sommige van de hier beschreven experimenten en tweaks je op weg door de jungle zullen helpen. Als je succes hebt of betere oplossingen vindt, horen we graag hoe het je vergaat; schrijf ons! ◀

210229-03

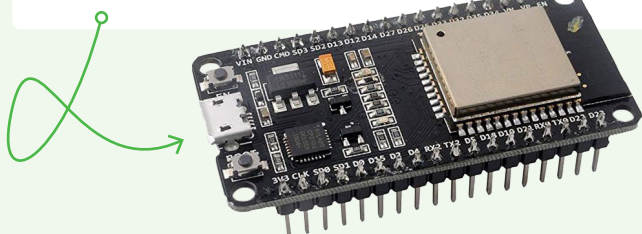
Vragen of opmerkingen?

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



Gerelateerde producten

- **ESP32-PICO-Kit V4 (with female headers) (SKU 20323)**
<https://elektor.nl/20323>
- **ESP32-DevKitC-32D (SKU 18701)**
<https://elektor.nl/18701>



WEBLINKS

- [1] De webpagina van dit artikel:
<https://elektormagazine.nl/210229-03>
- [2] ESP-IDF: <https://idf.espressif.com/>
- [3] Android Studio: <https://developer.android.com/studio>
- [4] Android Documentation: Request permission to access nearby Wi-Fi devices: <https://elektor.link/AndroidWiFiPermission>
- [5] Zen of Palm (2003) [PDF]: <https://cs.uml.edu/~fredm/courses/91.308-fall05/palm/zenofpalm.pdf>

elektor e-zine

Your dose of electronics



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

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



elektor
design > share > learn

Actief 1kHz-filter voor vervormingsmeting

meet beter door het het meetsignaal te optimaliseren

Alfred Rosenkränzer (Duitsland)

Het detecteren van een geringe signaalvervalsing stelt niet alleen hoge eisen aan het dynamisch bereik en de lineariteit van de meetapparatuur, maar ook aan het meetsignaal dat op de ingang van de te testen component wordt gezet. Met een achtergeschakeld filter kunnen echter zelfs 'gemiddelde' signaalgeneratoren hoogwaardige meetsignalen leveren.

In het artikel "Een Fliege-notchfilter voor audiometingen" in Elektor september/oktober 2022 [1] werd getoond hoe het meetbereik voor vervormingsmetingen met een digitizer kan worden vergroot door gebruik te maken van een notchfilter. Daarbij wordt het testsignaal onderdrukt, waardoor de digitizer zelf geen vervormingen kan veroorzaken. Een voorwaarde voor het correct detecteren van lage vervormingsniveaus van de apparatuur onder test (DUT) is natuurlijk dat de signaalgenerator die de DUT aanstuurt een kwalitatief hoorwaardig

signaal levert. Anders wordt naast de vervorming van het te testen apparaat ook de vervorming van de generator gemeten en worden onzuivere meetresultaten verkregen.

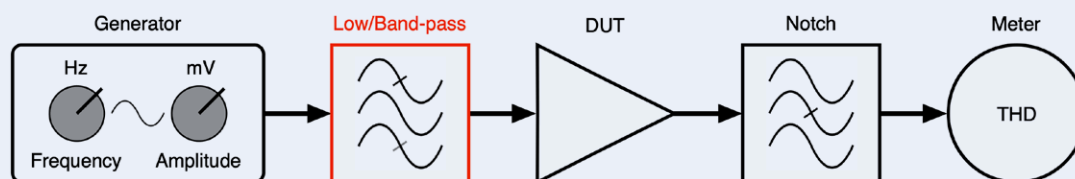
Bij betaalbare, niet-professionele signaalgeneratoren laat de kwaliteit van het gegenereerde signaal echter vaak te wensen over. Een achtergeschakelde filter kan hier een klein wonder verrichten en een bijzonder 'zuiver' meetsignaal opleveren door de vervormingen van de signaalgenerator te verzwakken.

Eisen

Aangezien de meeste audiometingen worden verricht met de typische meetfrequentie van 1 kHz, is een filter met een vaste doorlaatband of afsnijfrequentie bij deze waarde meestal voldoende. Het filter moet de testfrequentie onverzwakt doorlaten, maar de harmonischen (de veelvouden van de testfrequentie) zoveel mogelijk verzwakken. Een andere belangrijke eis is (uiteraard!) dat het filter zelf geen noemenswaardige vervorming mag veroorzaken.

Voor een eenvoudig gebruik is het handig als het filter zelf niet te smalbandig is, maar enige bandbreedte heeft. Op die manier kan de frequentie van het meetsignaal gemakkelijk zonder niveauveranderingen worden afgestemd op de exacte frequentie van het smalbandige notchfilter, zodat het filter het meetsignaal voor de meting verwijdert.

Figuur 1 toont het principe van zo'n meting met beide filters voor en na de DUT.



Figuur 1. Het geoptimaliseerde vervormings-meeprinpe met één filter voor en één na het DUT. Het voorgeschakelde laag- of banddoorlaatfilter (rood) verwijdert vervorming uit het generatorsignaal.

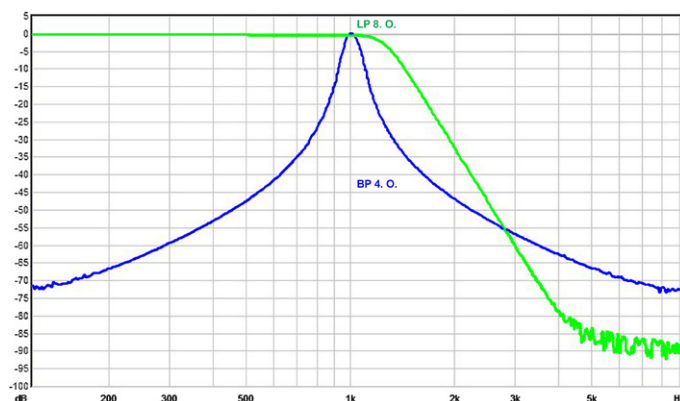
Filter

In principe kan de reductie van de vervorming van de signaalgenerator (harmonische verzwakking) niet alleen worden bereikt met een banddoorlaatfilter, maar ook met een passend laagdoorlaatfilter. De vraag is welke opstelling en welk filtertype de beste harmonische demping heeft in verhouding tot de kosten van de onderdelen. **Tabel 1** toont de gemeten verzwakkingen van een 4e-orde banddoorlaatfilter (BP, bandpass) in vergelijking met verschillende laagdoorlaatfilters (LP, lowpass) bij een testsignaal van 1 kHz. De demping bij 2 kHz en 3 kHz is bijzonder relevant, aangezien deze harmonischen bij de meeste generatoren de grootste amplitude hebben.

Tabel 1 Filterdemping

Frequentie (kHz)	BP 4e-orde (dB)	LP Bw. 4e-orde (dB)	LP Ch. 4e-orde (dB)	LP Bw. 8e-orde (dB)
1,0	0,0	0,1	0,3	0,0
2,0	-45,0	-11,7	-26,0	-32,0
3,0	-56,0	-25,2	-42,0	-60,0
4,0	-62,0	-35,0	-52,0	-80,0
5,0	-66,0	-42,8	-60,0	-90,0
6,0	-69,0	-49,5	-67,0	-90,0
7,0	-72,0	-54,0	-72,0	-90,0
8,0	-74,0	-59,0	-77,0	-90,0
9,0	-75,0	-63,0	-81,0	-90,0
10,0	-76,0	-67,0	-85,0	-90,0

Een Butterworth-laagdoorlaatfilter van de 4e orde haalt een verzwakking van iets minder dan 12 dB bij 2 kHz en is dus niet geschikt voor dit soort toepassing. Een Chebyshev-laagdoorlaatfilter van dezelfde orde

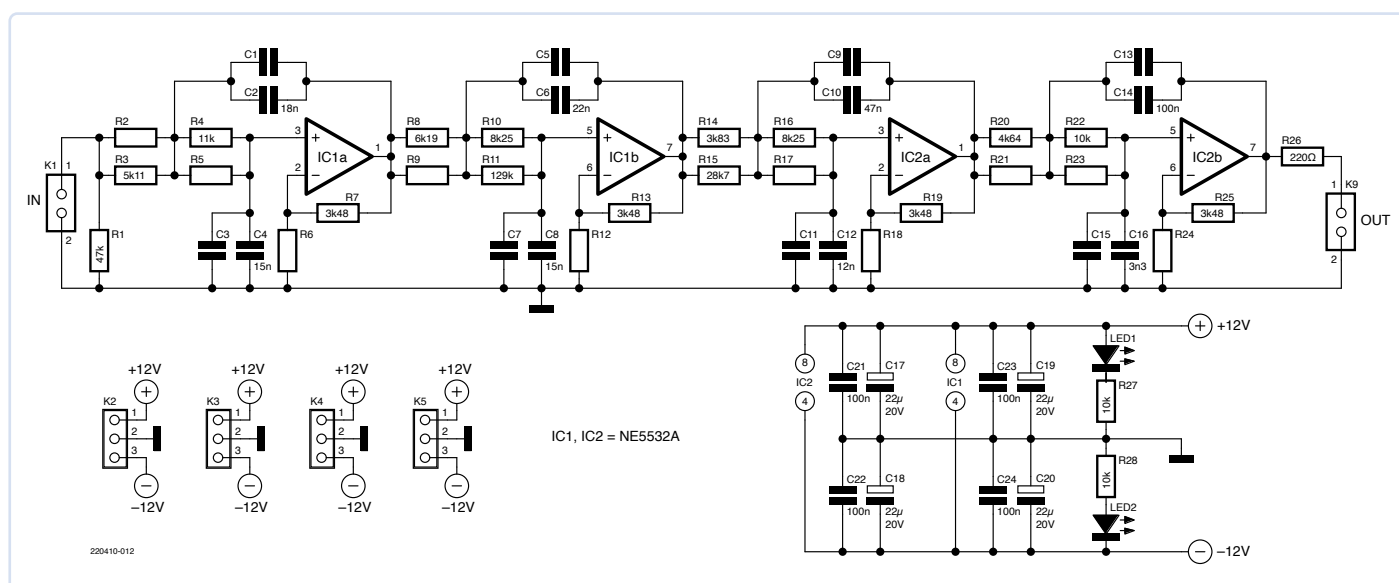


Figuur 2. Vergelijking van de frequentierespons van een 8e-orde Butterworth-laagdoorlaatfilter (groen) en een 4e-orde banddoorlaatfilter (blauw).

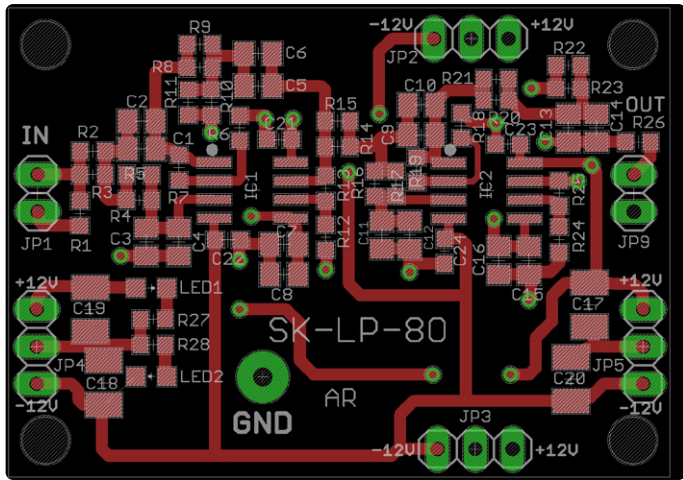
haalt hier al -26 dB. Een nog betere optie is een Butterworth-laagdoorlaatfilter van de 8e orde of een banddoorlaatfilter van de 4e orde met een bandbreedte van 100 Hz. **Figuur 2** toont een vergelijking van de frequentierespons van deze twee filters. Omdat deze twee filters de voorkeur verdienen, zijn er printplaten voor ontwikkeld met dezelfde afmetingen als het notchfilter uit [1]. De daarvoor ontwikkelde spanningsregelaar-print kan ook hier worden gebruikt.

De schakelingen

Figuur 3 toont de schakeling van het 8e-orde Butterworth-laagdoorlaatfilter. Om elke significante niveauverzwakking in een bereik van enkele tientallen hertz rond 1 kHz te vermijden, is de afsnijfrequentie ervan vastgelegd op 1,28 kHz met de gespecificeerde componentwaarden. **Figuur 4** toont de print-layout en **figuur 5** het prototype van het laagdoorlaatfilter. Voor de frequentiebepalende weerstanden en



Figuur 3. Schakeling van het 8e-orde Butterworth-laagdoorlaatfilter.



Figuur 4. Layout van de print voor het 8e-orde Butterworth-laagdoorlaatfilter.

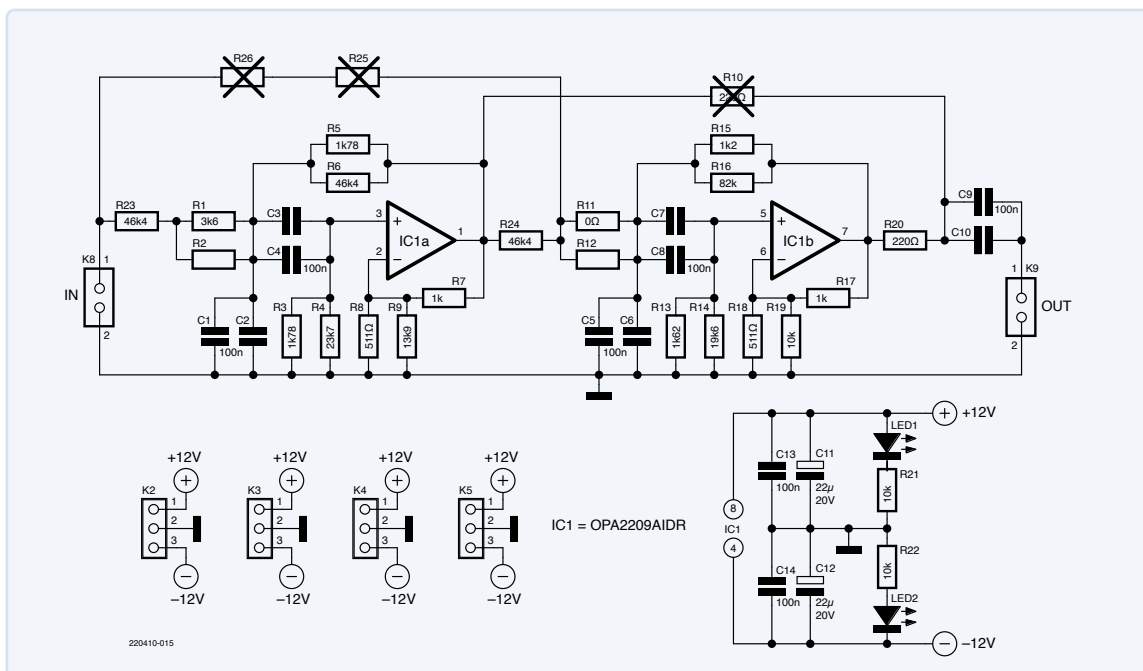


Figuur 5. Het voltooide laagdoorlaat-prototype van de auteur.

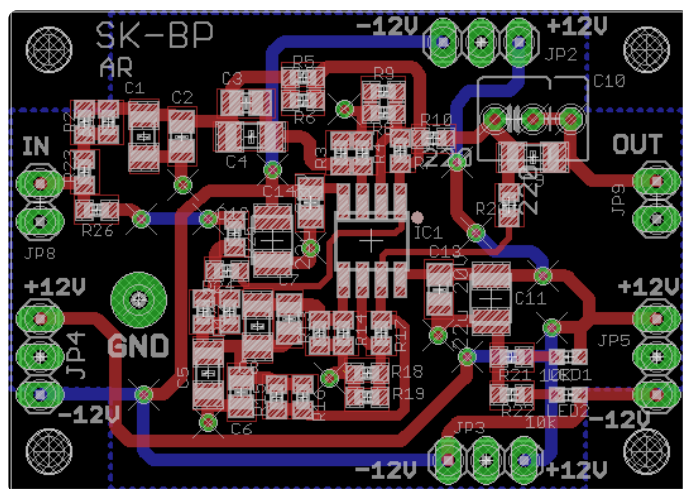
condensatoren is telkens een extra component parallel geschakeld (die zonder waarde in **figuur 3**), waardoor de ideale waarde zo nauwkeurig mogelijk kan worden benaderd met normale E-reeks componenten. Het is bijzonder raadzaam de condensatoren voor montage nauwkeurig op waarde te selecteren om te voorkomen dat later onderdelen vervangen moeten worden. Op de print is ook ruimte voor extra weerstanden (R6, R12, R18 en R24), die gebruikt kunnen worden voor de filterversterking, maar deze zijn hier niet nodig en worden daarom weggelaten. Het verdient in elk geval aanbeveling het artikel over het Fliege-notchfilter [1] te lezen en eventueel ook het artikel over filtersoftware [2], omdat daarin de procedure voor de exacte aanpassing van de filterfrequenties en de daarvoor gebruikte software om de componenten te berekenen uitvoerig worden toegelicht. Een optimalisatievoorbeeld: de gemeten (werkelijke) waarde van 100nF-condensator C14 is

103 nF, wat binnen de 5% tolerantie van de component valt. Als je nu een waarde van 103 nF in de software invoert, krijg je de aangepaste waarden voor de andere componenten van deze trap bij een bepaalde frequentie – en de juiste weerstanden zijn gemakkelijk en goedkoop verkrijgbaar, zelfs met een tolerantie van 1%.

Dezelfde procedure werd gevolgd voor het banddoorlaatfilter (schema: **figuur 6**, layout: **figuur 7**, prototype: **figuur 8**). Voor de frequentiebepalende componenten is ruimte voor parallelgeschakelde componenten gereserveerd; een optionele versterking werd gerealiseerd met R8, R9, R18 en R19. Aangezien de waarden van de frequentiebepalende condensatoren hier gelijk zijn, moet je indien mogelijk componenten uit dezelfde partij gebruiken, en de verschillen – die waarschijnlijk zeer klein zullen zijn – compenseren met parallelgeschakelde condensatoren. Ook hier wordt aanbevolen de resulterende waarde in de



Figuur 6. Schma van het 4e-orde banddoorlaatfilter met 100 Hz bandbreedte.



Figuur 7. Layout van de print voor het 4e-orde banddoorlaatfilter.

software in te voeren. Om de vereiste weerstandswaarden aan de ingang van elke trap zo nauwkeurig mogelijk te bereiken, werd telkens een combinatie van één weerstand in serie met een parallelschakeling van twee weerstanden toegepast. Weerstanden R10, R25 en R26 worden gebruikt om de eigenschappen van de afzonderlijke trappen te meten zoals beschreven in [1] en worden hier daarom weggelaten. Voor de fijnafstemming wordt aanbevolen de schakeling te simuleren met de werkelijk gemeten waarden.

Welk filter?

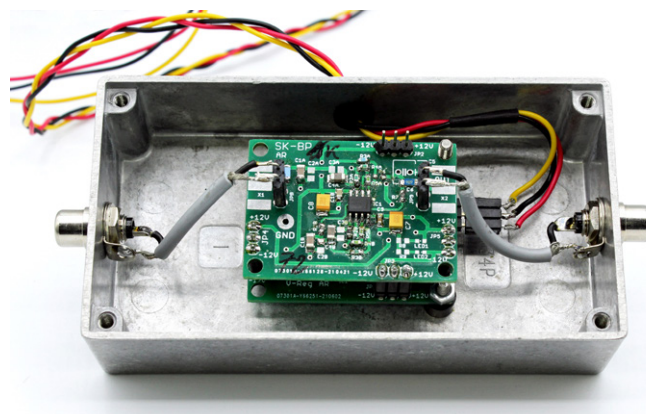
Nu je de keuze hebt tussen een laagdoorlaat- en een banddoorlaatfilter, rijst natuurlijk de vraag: welk filtertype is beter geschikt voor je doeleinden?

Het laagdoorlaatfilter moet van een hogere orde zijn om een bruikbare demping te leveren, en heeft daarom vier opamps nodig. Dit verhoogt de inherente ruis en natuurlijk de door het filter veroorzaakte vervorming. Anderzijds levert het een betere demping van hogere-orde harmonischen vanaf ongeveer 3 kHz dan het banddoorlaatfilter. Het feit dat het alle frequenties onder de afsnijfrequentie doorlaat, kan gunstig zijn voor andere toepassingen.

Het 4e-orde banddoorlaatfilter daarentegen heeft genoeg aan twee opamps en zou daarom minder eigen ruis en vervorming moeten produceren. Aangezien het alleen frequenties dicht bij 1 kHz doorlaat met een bandbreedte van 100 Hz, is het nauwelijks geschikt voor andere toepassingen. Maar aangezien hij alle frequenties onder 1 kHz verzwakt en dus ook ongewenste verschijnselen zoals netbrom, wordt ook de laagfrequente inherente ruis verminderd, wat de SNR-metingen ten goede kan komen.

Tot slot

Vanwege de kleine SMD-componenten die worden gebruikt, is het volbouwen van de printen niet geschikt voor mensen die voor het eerst met de hand solderen. Ervaren elektronica-makers zouden er echter geen problemen mee mogen hebben. De layout-bestanden kunnen gratis worden gedownload (in Eagle-formaat) van de webpagina bij dit artikel [3]. Vanwege hun eigenschappen moeten voor alle weerstanden in de signaalweg hoogwaardige dunne-film versies worden gebruikt. De signaalvoerende keramische SMD-condensatoren moeten beslist



Figuur 8. Het prototype van het banddoorlaatfilter, gemonteerd in een aluminium behuizing.

types zijn met een COG-dielectricum om stabiele eigenschappen en een geringe inherente vervorming te garanderen. De auteur heeft nog enkele lege printen over. Indien gewenst kun je per e-mail contact met hem opnemen (zie het kader **Over de auteur**).

De twee hier gepresenteerde filters zijn natuurlijk niet beperkt tot 1 kHz. Je kunt gemakkelijk een hele reeks geschikte laag-/banddoorlaat- en notchfilters bouwen voor verschillende frequenties, onder toepassing van de ontwerpoverwegingen in [1], en ze gebruiken om het vervormingsgedrag van audioschakelingen bij verschillende frequenties te onderzoeken. ◀

220410-03

Over de auteur

Alfred Rosenkränzer werkte vele jaren als ontwikkelingsingenieur, aanvankelijk op het gebied van professionele televisietechnologie. Sinds het einde van de jaren negentig ontwikkelt hij digitale high speed- en analoge schakelingen voor IC-testers. Audio is zijn privé-liefhebberij.

Vragen of opmerkingen?

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

WEBLINKS

- [1] Alfred Rosenkränzer, "Een Fliege-notchfilter voor audiometingen", Elektor september/oktober 2022: <https://www.elektormagazine.nl/magazine/elektor-274/60995>
- [2] Alfred Rosenkränzer, "Filtersoftware", Elektor november/december 2022: <https://www.elektormagazine.nl/magazine/elektor-282/61123>
- [3] Webpagina bij dit artikel: <https://www.elektormagazine.nl/220410-03>



ONDERDELENLIJST LAAGDOORLAATFILTER

Weerstanden (dunnefilm, 1%, SMD 0603):

R1 = 47 k
R2, R5, R9, R17, R21, R23 = voor parallelmontage *
R3 = 5k11
R4 = 11 k
R6, R12, R18, R24 = versterking, niet monteren *
R7, R13, R19, R25 = 3k48
R8 = 6k19
R10, R16 = 8k25
R11 = 129 k
R14 = 3k83
R15 = 28k7
R20 = 4k64

R22, R27, R28 = 10 k

R26 = 220 Ω

Condensatoren

(tenzij anders vermeld: 5%, COG, SMD0805):

C1, C3, C5, C7, C9, C11, C13, C15 =
voor parallelmontage *
C2 = 18 n
C4, C8 = 15 n
C6 = 22 n
C10 = 47 n
C12 = 12 n
C14 = 100 n

C16 = 3n3

C17, C18, C19, C20 = elco 22 μ /20 V, SMD-B
C21, C22, C23, C24 = 100 n, 0603

Halfgeleiders:

LED1, LED2 = LED, groen, SMD0805
IC1, IC2 = NE5532A, SO08

Overig:

K1, K9 = 2-polige header male, raster 0,1"
K2...K5 = 3-polige header male, raster 0,1"

* zie tekst



ONDERDELENLIJST BANDDOORLAATFILTER

Weerstanden (dunnefilm, 1%, SMD 0603):

R1 = 3k6
R2, R12 = voor parallelmontage *
R3, R5 = 1k78
R4 = 23k7
R6, R23, R24 = 46k4
R7, R17 = 1 k
R8, R18 = 511 Ω
R9 = 13k9
R10, R25, R26 = niet monteren *
R11 = 0 Ω

R13 = 1k62

R14 = 19k6

R15 = 1k2

R16 = 82 k

R19, R21, R22 = 10 k

Condensatoren

(tenzij anders vermeld: 5%, COG, SMD0805):

C1, C4, C5, C8, C9 = 100 n, 1206
C2, C3, C6, C7 = voor parallelmontage *
C10 = niet monteren

Halfgeleiders:

LED1, LED2 = LED, groen, 0603
IC1 = OPA2209AIDR, SO08

Overig:

K8, K9 = 2-polige header male, raster 0,1"
K2...K5 = 3-polige header male, raster 0,1"

* zie tekst



Gerelateerde producten

- > OWON AG051 Arbitrary Waveform Generator (5 MHz) (SKU 18874)
<https://elektor.nl/18874>
- > OWON SDS1102 2-Channel Oscilloscope (100 MHz) (SKU 18782)
<https://elektor.nl/18782>
- > OWON XSA810 Spectrum Analyzer (1 GHz) (SKU 19714)
<https://elektor.nl/19714>



Alle begin...

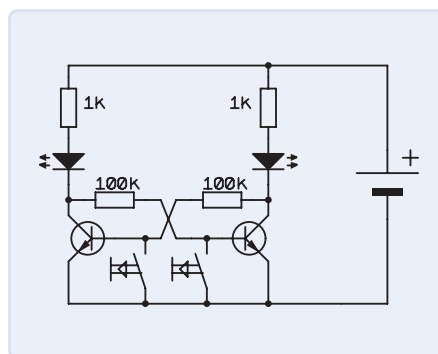
...multivibreert verder!

Eric Bogers (Elektor)

In de vorige aflevering hebben we (eindelijk) kennis gemaakt met de transistor – een component die we tegenwoordig gerust het ‘hart van onze samenleving’ mogen noemen: zonder transistoren in de een of andere vorm zou onze samenleving in de huidige vorm onmogelijk zijn. Nu gaan we verder met multivibratoren en bekijken we de transistor als versterker.

Eerst een kort woord vooraf, ten behoeve van vooral nieuwe Elektor-lezers die voor het eerst een aflevering van de reeks “Alle begin...” onder ogen krijgen. Deze artikelserie pretendeert op geen enkele manier een gedegen theoretische cursus in de elektronica te zijn. Het enige wat we willen is de beginnende elektronica-liefhebber voldoende weetjes aan te reiken zodat hij of zij met redelijke kans op succes zelf een schakeling kan ontwerpen en bouwen (of een bestaande schakeling kan modificeren). Daarbij gaan we met een grote boog om het betere rekenwerk heen, en nemen we het met de theorie hier en daar ook niet al te nauw. Het is ons duidelijk dat we op die manier de perfectionisten en theoretici onder onze lezers af en toe grandioos op de tenen zullen trappen – dat spijt ons dan een beetje, maar ook weer niet al te veel... De vorige keer hebben we als laatste de bistabiele multivibrator of flipflop besproken; in **figuur 1** drukken we het prinsipschema nogmaals af. Deze schakeling dankt zijn naam aan het feit dat hij twee stabiele

toestanden kent: ofwel de ene ofwel de andere transistor is in geleiding. Omschakelen is mogelijk met behulp van de druktoetsen. Welke van de beide transistoren direct na het inschakelen van de voedingsspanning in geleiding zal zijn, valt onmogelijk te voorspellen; de enige zekerheid die we hebben is dat één van beide zal geleiden en de andere zal sperren.



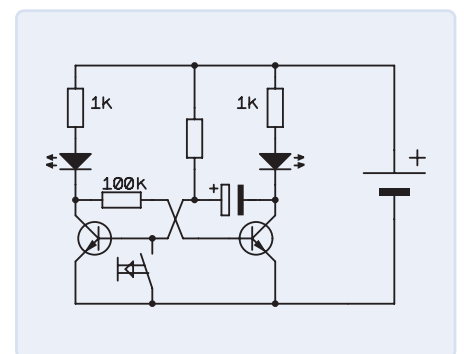
Figuur 1. De bistabiele multivibrator of flipflop.

De monostabiele multivibrator

Een monostabiele multivibrator (ook bekend onder de koosnaam *monoflop*) is afgeleid van zijn bistabiele broertje, en kent (zoals de naam al suggereert) slechts één stabiele toestand. Het prinsipschema is geschetst in **figuur 2**. De schakeling kan door een impuls ‘van buiten’ weliswaar in een andere toestand worden gebracht, maar zal na verloop van een bepaalde tijd vanzelf weer in de stabiele (uitgangs) toestand terugkeren. Misschien ken u dit soort schakelingen van die handige trappenhuisverlichting, die zich na verloop van tijd vanzelf weer uitschakelt.

In de schakeling van figuur 2 is de linker transistor in eerste instantie in geleiding. Net als bij de flipflop (figuur 1) kunnen we deze transistor doen sperren door zijn basis via de drukknop naar massa kort te sluiten. Daardoor wordt ook meteen de elco ontladen.

Nu komt de rechter transistor (tijdelijk) in geleiding. De elco wordt dan echter ook weer (via diens weerstand) opgeladen. Zodra de spanning over de elco hoger wordt dan



Figuur 2. De monostabiele multivibrator.

ongeveer 0,7 V, komt de linker transistor weer in geleiding zodat diens CE-spanning tot vrijwel 0 V afneemt. Dat heeft tot gevolg dat de rechter transistor weer gaat sperren. De stabiele toestand is weer bereikt.

De tijd die verstrijkt voordat de monoflop weer omklapt naar zijn stabiele toestand, hangt (uiteraard) af van de waarde van de (laad)weerstand en de condensator, maar ook van de hoogte van de voedingsspanning. Laten we eens een poging wagen deze tijd (de 'monotijd') 'uit het losse polsje' en met de natte vinger te berekenen. We gaan uit van de volgende waarden: $R = 100 \text{ k}\Omega$, $C = 100 \text{ }\mu\text{F}$ en $U_V = 12 \text{ V}$. Voor de spanning over een condensator geldt:

$$U_C = \frac{Q}{C} = \frac{I \cdot t}{C}$$

De stroom wordt bepaald door de spanning over de weerstand en de waarde van die weerstand. Deze spanning neemt continu af naarmate de condensatorspanning toeneemt. Omdat de schakeling bij een condensatorspanning van 0,7 V al omklapt, verwaarlozen we het niet-lineaire gedrag en rekenen we met het rekenkundige gemiddelde.

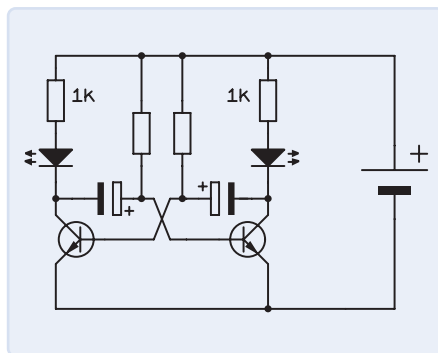
$$U_C = \frac{U_V \cdot t}{R \cdot C} \Rightarrow t = \frac{U_C \cdot R \cdot C}{U_V}$$

De schakeling zal bij ongeveer 0,7 V omklappen; dat betekent dat de gemiddelde spanning over de weerstand zo ongeveer 11,65 V bedraagt.

$$t = \frac{0,7 \text{ V} \cdot 100 \text{ k}\Omega \cdot 100 \text{ }\mu\text{F}}{11,65 \text{ V}} = 0,6 \text{ s}$$

De astabiele multivibrator

Een astabiele multivibrator kent geen stabiele toestand, maar schakelt voortdurend tussen de beide mogelijke toestanden heen en weer – daarom ook de naam van het beestje. De schakeling van **figuur 3** ziet er zo'n beetje uit als een dubbele monoflop, zodat we geneigd zouden zijn voor de timing de bij de monoflop afgeleide formules te gebruiken – maar daarmee zouden



Figuur 3. De astabiele multivibrator.

we er ongeveer een factor 10 naast zitten. Dat geeft natuurlijk aanleiding tot de vraag hoe deze schakeling dan eigenlijk werkt. Laten we aannemen dat de linker transistor geleidt: U_{CE} van de linker transistor is dan ongeveer nul en die van de rechter transistor ongeveer gelijk aan de voedingsspanning. De linker condensator wordt via zijn laadweerstand langzaam opgeladen; op een gegeven moment bedraagt de spanning over de condensator 0,7 V en de rechter transistor schakelt door.

Bij het doorschakelen (in geleiding komen) van de rechter transistor is de verandering van U_{CE} ongeveer gelijk aan de hoogte van de voedingsspanning, en dat betekent dat de condensatorpotentiaal met deze hoeveelheid afneemt: de basis van de linker transistor voert zodoende een negatieve potentiaal en de transistor spert dan ook. Maar dat betekent dat de condensator niet vanaf nul moet worden opgeladen, maar vanaf een spanning die ongeveer even groot is als de voedingsspanning, maar dan met een minteken. En dat neemt uiteraard veel meer tijd in beslag.

Laten we hier eens aan gaan rekenen. De condensator moet dus van -12 V tot $+0,7 \text{ V}$ worden opgeladen. Als we daarvoor een lineaire laadcurve zouden aannemen, introduceren we een fout van ongeveer 10% (ongeveer evenveel als de tolerantie van een elco van redelijke kwaliteit). Voor de aardigheid proberen we hier wat nauwkeuriger te zijn. Voor de lading van een condensator geldt:

$$U_C = U_0 \cdot \left(1 - e^{\frac{-t}{RC}} \right)$$

Nu is in dit geval U_0 gelijk aan de dubbele voedingsspanning, en de condensator wordt opgeladen tot $U_V + 0,7 \text{ V}$. Zodoende kunnen we schrijven:

$$U_C + U_V = 2 \cdot U_V \cdot \left(1 - e^{\frac{-t}{RC}} \right)$$

Uit deze vergelijking kunnen we t oplossen; met de afleiding zullen we u niet lastigvalen, we geven hier alleen het resultaat:

$$t = -R \cdot C \cdot \ln \left(1 - \frac{U_C + U_V}{2 \cdot U_V} \right)$$

Wanneer we dit uitrekenen met dezelfde onderdelenwaarden als bij de monoflop ($R = 100 \text{ k}\Omega$ en $C = 100 \text{ }\mu\text{F}$), komen we uit op $t = 7,5 \text{ s}$, en dat is 12 keer zo lang als bij de monoflop.

De bipolaire transistor als versterker

Een bipolaire transistor is eigenlijk geen bijzonder 'dankbaar' onderdeel om een versterker mee te construeren: alle parameters hangen onderling van elkaar af, geen enkele van die parameters is echt lineair en alles is bovendien op een 'kwalijske' manier van de temperatuur afhankelijk – als geen tegenmaatregelen worden getroffen, dan vliegt om te beginnen het werkpunt uit de bocht.

Om deze redenen worden voor kleinsignaaltoepassingen nog maar zelden losse transistoren ingezet: we gebruiken bij voorkeur operationele versterkers. Die zijn weliswaar ook met transistoren opgebouwd, maar bij opamps zijn de meeste problemen door de ontwerpers al gecompenseerd, zodat we daar op bijzonder eenvoudige manier stabiele versterkers mee kunnen bouwen.

Operationele versterkers zijn echter eigenlijk alleen voor kleinsignaalbedrijf geschikt: voor vermogensversterkers zijn de maximale uitgangsspanning en -stroom doorgaans te gering. Opamps kunnen in zo'n geval worden 'opgevoerd' met discrete (vermogens)transistoren – of we kunnen de hele schakeling natuurlijk van meet af aan met losse transistoren opbouwen.

Opamps komen in een volgende aflevering aan de orde. Om die dan beter te kunnen begrijpen, houden we ons hier kort bezig met de transistor als versterkende component.

In de meeste leerboeken volgen op dit punt een heleboel theorie en vergelijkingen die echter niet bijzonder veel met de praktijk te maken hebben. De enig werkelijk belangrijke relatie is de U/I-karakteristiek van de basis/emitter-diode, en die zijn we bij de diode al tegengekomen – zie **figuur 4**. Bij de besprking hebben we toen echter verzwegen dat de U/I-karakteristiek temperatuurafhankelijk is. Als de diode warm wordt, dan neemt (bij gelijkblijvende spanning) de stroom toe (waardoor de diode nog warmer wordt). Dit gedrag levert bij een gewone diode zelden problemen op, maar bij een transistor is het bijzonder kwalijk omdat daardoor ook de collectorstroom zal variëren – en wel ongeveer een factor 100 sterker. Voor de collectorstroom geldt namelijk:

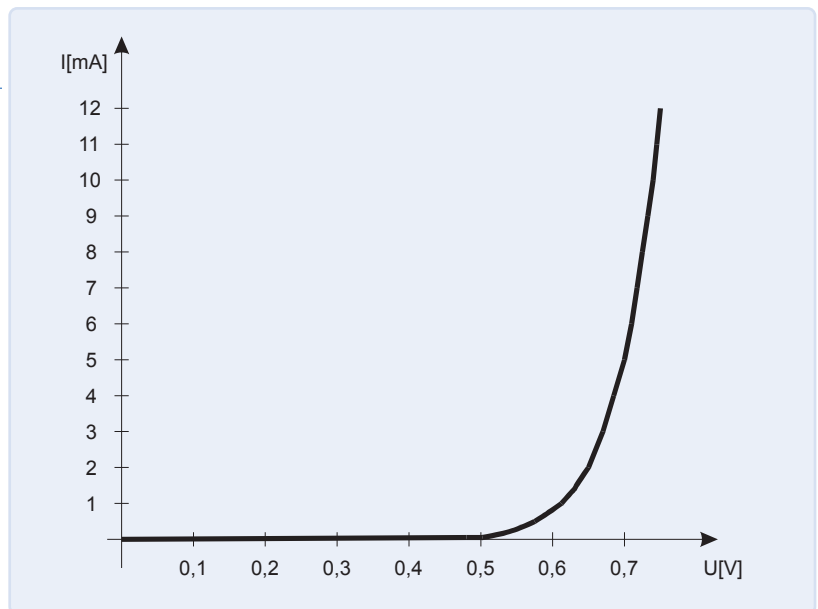
$$I_C = I_B \cdot h_{FE} = I_B \cdot \beta$$

Met h_{FE} of β wordt de stroomversterking van de transistor bedoeld. Deze stroomversterking is echter niet alleen onderhevig aan grote exemplarische spreiding, maar is (om het nog erger te maken) ook nog eens afhankelijk van de collectorstroom. Bij kleinsignaaltransistoren ligt de waarde van h_{FE} ergens tussen 100 en

800, en bij vermogenstransistoren tussen 10 en 100. Omdat de stroomversterking zo'n onbetrouwbare parameter is, moeten we onze transistorschakelingen zo proberen te ontwerpen dat de exacte waarde van h_{FE} geen (noemenswaardige) rol van betekenis speelt. Dat zullen we in de volgende aflevering zien. ◀

230033-03

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



Figuur 4. De U/I-karakteristiek van een diode.

Vragen of opmerkingen?

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



Gerelateerde producten

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

YOUR KEY TO CELLULAR TECHNOLOGY



**WURTH
ELEKTRONIK**
MORE THAN
YOU EXPECT

**WE meet @
PCIM Europe**
Hall 6-217

Adrastea-I is a Cellular Module with High Performance, Ultra-Low Power Consumption, Multi-Band LTE-M and NB-IoT Module.

Despite its compact size, the module has integrated GNSS, integrated ARM Cortex M4 and 1MB Flash reserved for user application development. The module is based on the high-performance Sony Altair ALT1250 chipset. The Adrastea-I module, certified by Deutsche Telekom, enables rapid integration into end products without additional industry-specific certification (GCF) or operator approval. Provided that a Deutsche Telekom IoT connectivity (SIM card) is used. For all other operators the module offers the industry-specific certification (GCF) already.

www.we-online.com/gocellular

- Small form factor
- Long range/worldwide coverage
- Security and encryption
- Multi-band support

#GOCCELLULAR

Project 2.0

correcties, updates en brieven van lezers

samengesteld door Jens Nickel (Elektor)



Landbouw in zakformaat

Elektor Arduino-editie 2022, pag. 54 (220414)

Met belangstelling heb ik uw artikel in Elektor gelezen. Ik heb in het verleden zaden geplant, maar was niet erg succesvol omdat het me niet lukte de zaden de liefdevolle zorg te geven die ze nodig hebben. Uiteindelijk heb ik mijn desktop-kassen aan mijn vriendin geschonken, die meer van planten houdt en beter voor ze zorgt. Hoe dan ook, ik vond uw artikel erg leuk en wil iets soortgelijks bouwen voor mijn vriendin. U vermeldt dat planten de voorkeur geven aan licht met golflengtes 450 nm en 650 nm. RGB-LED's produceren golflengtes van 460, 525 en 625 nm (snel met Google opgezocht). Deze waarden komen niet helemaal overeen met de voorkeuren van de planten. Zijn ze toch goed genoeg? Elektor publiceerde in 2019 samen met Würth Elektronik al een artikel over een kas met LED-verlichting (www.elektormagazine.nl/magazine/elektor-112/51089). Uw artikel inspireerde me om me in dit onderwerp te verdiepen. Ik heb hier al jaren een ultrasone vernevelaar liggen, dus die kan eindelijk van pas komen.

Michael Hompus

Bedankt voor je vraag! Enige tijd geleden, nadat ik het kweekbox-project online had gepubliceerd, begon ik onderzoek te doen naar het onderwerp van LED-lampen voor plantengroei, en je bent van harte welkom om mijn samenvatting daarover hier te lezen:

instructables.com/DIY-Grow-LED-Light-Designing-a-Better-Sun

Om je vraag te beantwoorden en je te helpen de nieuwste knowhow op dit gebied praktisch toe te passen: gebruik witte kleuren-LED's plus rode of ver-rode spectrum-LED's. Het idee is om een volledig spectrum na te bootsen dat vergelijkbaar is met dat van de zon, en vervolgens de efficiëntie te verhogen door wat extra rode/ver-rode/UV-LED's toe te voegen. Voor wit zou ik Samsung LM301B- of LM301H-diodes aanbevelen. Voor rood de hyperrode LED's van Osram. Praktisch ziet het er ongeveer zo uit: 6...10 witte LED's plus twee rode LED's op een fyto-lichtbalk. Dan speel je gewoon een beetje met de kleurtemperatuur van de witte LED's – als je koud wit kiest, krijg je meer blauw in het spectrum, als je warm wit kiest, krijg je meer rood in het spectrum, enzovoort. Ik hoop dat het basisidee duidelijk is.

Bezoek cityfarm.md om de definitieve versie van het kweekbox-project te zien.

Dmitrii Albot (auteur van het artikel)



Goedkope audiotester

Elektor juli/augustus 2022, p. 6 (200604)

In de DHZ-community wordt de REW-software (roomeqwizard.com) veel toegepast. REW is in principe freeware, hoewel je een donatie aan de maker kunt (of hoort te) doen. Deze software zou alles moeten kunnen doen wat in het artikel wordt genoemd. Voor degenen met aanzienlijk hogere eisen is er ook een Pro-versie, maar deze kost 100 dollar.

Er staat een lange thread over REW op diyaudio.com, die ook verschillende interfaces en de modificatie daarvan behandelt.

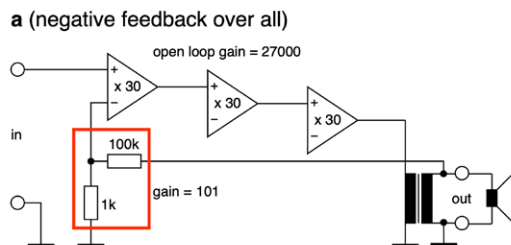
Frank von Zeppelin



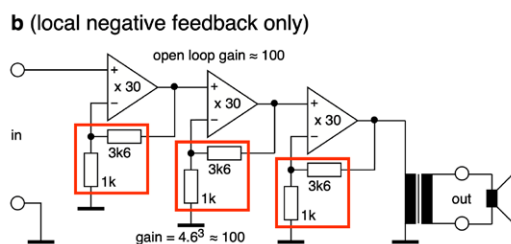
The Tube

Elektor januari/februari 2023, p. 70 (220089)

Ik ben al jaren abonnee. Mijn eerste Elektor-project was een FM-tuner met printspoelen in 1975, toen ik ongeveer 20 jaar oud was. Ik ben blij dat u gedurende vele decennia uw tijdschriftconcept en al het bijbehorende aanbod hebt aangepast aan de werkelijke behoeften van mensen. Ga zo door! In het laatste nummer heb ik me echter verwonderd over sommige zaken in het artikel "The Tube" – misschien zijn het gewoon typefouten? Of misschien heb ik iets verkeerd begrepen?



Figuur 1a: De uitleg in de tekst suggereert (indirect) dat er geen tegenkoppeling wordt toegepast in de afzonderlijke trappen (bijvoorbeeld door kathodeweerstand). Ik kan me moeilijk voorstellen dat deze aanpak echt mogelijk is. Anders had de auteur toch op dit exotische speciale geval moeten wijzen?



Figuur 1b: Met de vermelde weerstandswaarden zou de totale versterking eigenlijk 4,3 tot de derde macht (en dus 79,5) moeten zijn, toch? (Bovendien is 3,3 tot de derde macht ongeveer 36, wat niet in de buurt komt van 100.) En is het eigenlijk mogelijk om in deze situatie te praten over open-lus versterking? Ik denk dat "overall-versterking van ingang naar uitgang" een betere term zou zijn.

Harald Sonneman

Je hebt helemaal gelijk dat geen levensvatbare audioversterker het zonder enige tegenkoppeling kan stellen, omdat de versterkende componenten niet lineair genoeg zijn. Buizenliefhebbers gebruiken de term "zonder tegenkoppeling" vaak als een soort synoniem voor "zonder overall-tegenkoppeling".

Wat ik eigenlijk wilde was overall-tegenkoppeling tegenover het principe van lokale tegenkoppeling per versterkingstrap stellen; zie **figuur 1** in het artikel.

In het heetst van de strijd heb ik me echter inderdaad verrekend. Zoals je opmerkt is 4,3 tot de derde macht 79, niet 100, en 3,3 tot de derde macht is al helemaal geen 100. Ik heb deze fout niet opgemerkt bij het proeflezen van de tekst, en de auteur ook niet toen hij het geredigeerde artikel goedkeurde. We drukken hier een herzien schema af.

En natuurlijk betekent "open-lus versterking" in figuur 1b eigenlijk 'quasi open-lus versterking'.

Thomas Scherer (redacteur van het artikel)

Alle begin...

Elektor september/oktober 2022, p. 34 (220256)

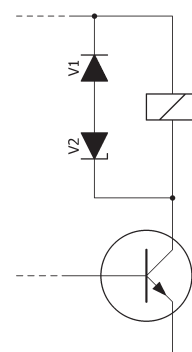
Project 2.0

Elektor januari/februari 2023, p. 110 (220601)

Helaas is er een fout geslopen in mijn suggestie, gepubliceerd als lezersbrief in het januari/februari-nummer 2023.

Ik heb daar voorgesteld om bij het aansturen van een relais met een transistor een zenerdiode in serie te zetten met de vrijlooptiode. Dat verkort niet alleen de uitschakeltijd van het relais aanzienlijk, maar helpt ook de levensduur van de relaiscontacten te verlengen, omdat de contacten aanzienlijk sneller openen.

Helaas werd in de bijbehorende tekening de zenerdiode verkeerd-om getekend, met als gevolg dat alleen de doorlaatspanningen van de twee diodes worden opgeteld – en daar schiet je niet veel mee op. In de bijgaande figuur hier is de zenerdiode in de juiste richting aangesloten.



Wanneer de transistor uitschakelt, wordt de sperspanning V_2 van de zenerdiode opgeteld bij de doorlaatspanning V_1 van de gelijkrichtdiode. Hierdoor neemt de stroom door het relais veel sneller af. Dit is niet alleen meetbaar, maar ook duidelijk hoorbaar.

In dit opzicht is het belangrijk om te bedenken dat de spanning tussen de collector en de emitter van de transistor de som is van de voedingsspanning, de sperspanning van de zenerdiode en de doorlaatspanning van de gelijkrichtdiode. Bij een voedingsspanning van 12 V en een zenerdiode van 10 V ziet de transistor op het moment van uitschakelen een spanning van ongeveer 22,6 V, waarmee rekening moet worden gehouden bij de keuze van de transistor en de zenerdiode.

Thomas Klingbeil

Het nieuwe I3C-protocol

een waardig opvolger van I²C – of gebakken lucht?

Tam Hanna (Slowakije)

Er is vrijwel geen busprotocol dat zo veel gebruikt wordt als het I²C-protocol dat begin jaren tachtig werd geïntroduceerd door Philips Semiconductor. Maar de laatste jaren komen de zwakke punten van dit bussysteem steeds duidelijker aan het licht. Er bestaat een opvolger in de vorm van het I3C-protocol, en in dit artikel willen we dat in detail bekijken.

Om de mogelijkheden en de beperkingen goed te begrijpen, moet je weten dat de I²C-bus tegenwoordig een rechtenvrije technologie is, nu de patenten van Philips verlopen zijn. Hoewel NXP, als opvolger van de halfgeleiderdivisie van Philips, nog steeds eigenaar is van de intellectuele rechten op het logo en de naam, kan de technologie nu door iedereen gratis worden gebruikt.

De discussie over I²C versus TWI is vrijwel verstomd, maar er zijn online nog steeds interessante uitspraken te vinden (bijvoorbeeld bij [1]) met de volgende teneur: "Het idee achter het betalen van licentierechten was om zeker te stellen dat mensen die zeiden dat ze I²C-technologie maakten (en die als I²C verkochten) de standaard ondersteunden en zo meehielpen die te laten groeien."

Het resultaat was dat Philips de licentiehouders altijd aanmoedigde om de I²C-standaard volledig te implementeren. Daar hoorde het glitchfilter expliciet bij, wat in het verleden vaak conflicten opleverde tussen Philips Semiconductor en andere fabrikanten. Dit filter (we komen er later nog op terug) was misschien wel één van de redenen waarom Philips licentiebetalen ging verlangen.

Level Annual Turnover	1st Quarter Fees (Jan, Feb, Mar)	Annual Fee
Adopter > USD \$250m	\$8,000	\$8,000
Adopter ¹ < USD \$250m	\$4,000	\$4,000
Contributor > USD \$250m	\$32,000	\$32,000
Contributor ¹ < USD \$250m	\$16,000	\$16,000
Contributor ² < USD \$10m	\$8,000	\$8,000
Board (and Promoter)	\$40,000	\$40,000

Figuur 1. Het MIPI-lidmaatschap is een kostbare zaak (bron: [3]).

Maar nu de patenten van Philips zijn verlopen, is het jachtseizoen geopend.

Helaas kan de werking van I²C-componenten in een I3C-bussysteem hooguit worden gezien als ofwel *best practice* of *wishful thinking*. Praktijktests zijn altijd nodig bij zulke gemengde systemen. Maar laten we eerst eens kijken wie of wat er achter de I3C-standaard zit.

I3C en de MIPI Alliance

De *MIPI Alliance* [2], een comité dat standaarden voor allerlei elektronica uitbrengt, is verantwoordelijk voor de I3C-standaard. De lidmaatschapsgelden voor de alliantie zijn niet gering, zoals blijkt uit de verborgen prijslijst (**figuur 1**) [3]. Om de adoptie te stimuleren biedt het standaardiseringscomité een volledig gratis basisversie van de I3C-specificatie aan.

Een lijst van geavanceerde functies (alleen beschikbaar voor leden van de MIPI Alliance) is te vinden bij [4]. Op die pagina is ook te zien dat I3C staat voor *Improved Inter Integrated Circuits*. De auteur denkt dat als je I3C gebruikt in een commerciële schakeling, er alleen 'gedoe' met de MIPI Alliance kan ontstaan als je de interface zelf gaat implementeren, bijvoorbeeld in een FPGA of door bit-banging. Maar als je kiest voor kant-en-klare componenten (en het I3C-logo of de naam niet gebruikt in advertenties), zijn er geen problemen te verwachten.

Vaarwel open drain

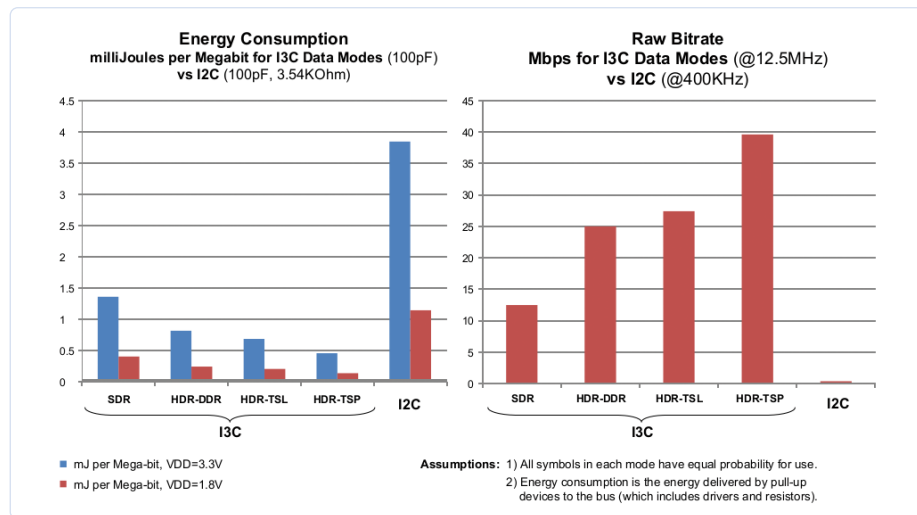
I²C-bus-ontwikkelaars zijn van meet af aan geplaagd door kleine en grote irritaties. Om te beginnen is er altijd de vraag waar pull up-weerstand (voor het ontladen van de buscapaciteit) moeten worden geplaatst. Verder is het meestal niet mogelijk om interrupts te triggeren vanuit een component die niet zelf het kloksignaal genereert. Als je een 'clock sensing'-component interrupts wilt laten genereren, is altijd een extra I²C-verbinding nodig. Als je meerdere sensoren in het veld hebt, gaat dat ten koste van de GPIO-capaciteit van de component die het kloksignaal genereert.

De derde irritatie is de vrij lage snelheid: Philips specificeert een overdrachtssnelheid van 3 Mbps bij een werkfrequentie van 3,4 MHz. Het gevolg van deze traagheid is een groter energieverbruik, want hoe langer de data-overdracht tussen twee componenten duurt, des te langer het ook duurt voordat ze naar een energiezuinige rusttoestand overgaan. Het MIPI-diagram (**figuur 2**) maakt dit zonder meer aannemelijk.

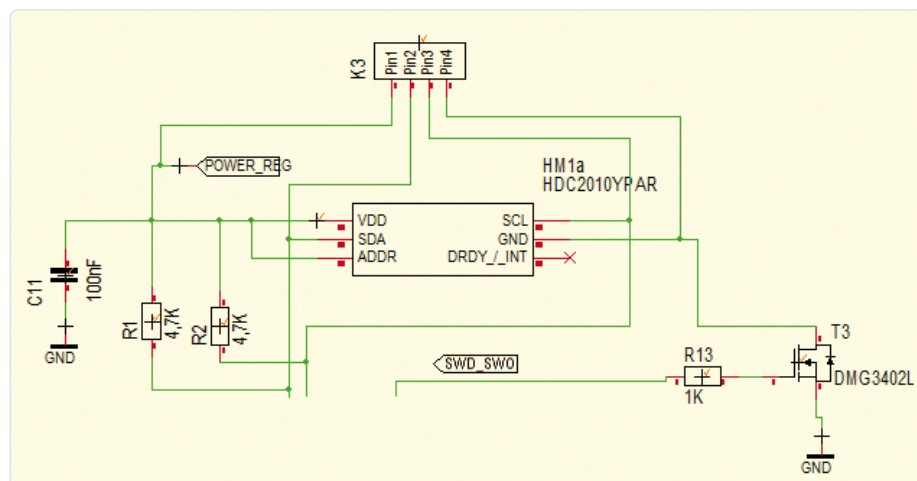
En tenslotte: I²C-componenten kunnen andere componenten op de bus 'voor eeuwig' buiten-sluiten. Een schemadetail zoals in **figuur 3** is in veel I²C-systemen te vinden.

Verwelkom de I3C-verbeteringen

De eerste verandering die I3C met zich meebrengt, is dat zowel SCL als SDA meestal in push/pull-modus werken. De SCL-pin werkt altijd in push/pull, de SDA-pin werkt onder bepaalde omstandigheden nog wel als open-drain. Om verschillende problemen bij de plaatsing van de pull up-weerstanden te voorkomen, schrijft de standaard voor dat die zich altijd in de component moeten bevinden die de klok genereert.



Figuur 2. Snelheid spaart energie, net als bij mobiele processoren (bron: [8]).



Figuur 3. 'Noodrem'-transistor T3 schakelt alle op de bus aangesloten sensoren 'hard' uit om een blokkerende component weer in de default-toestand te brengen.

Een ander belangrijk punt is dat SCL altijd (afgezien van enkele bijzondere gevallen) wordt bestuurd door de component die het kloksignaal genereert. Als er op een bus meerdere klok-genererende componenten zijn aangesloten, is de component dat op dat moment de klok genereert verantwoordelijk voor het schakelen van de SCL-pin.

Bij een conventionele I²C-bus kan een slave-IC het kloksignaal pauzeren. In de I3C-specificatie wordt dat niet ondersteund. Zulke componenten kunnen daarom op een gemengde bus niet functioneren. In de praktijk heb ik weinig praktische toepassingen van deze techniek gezien.

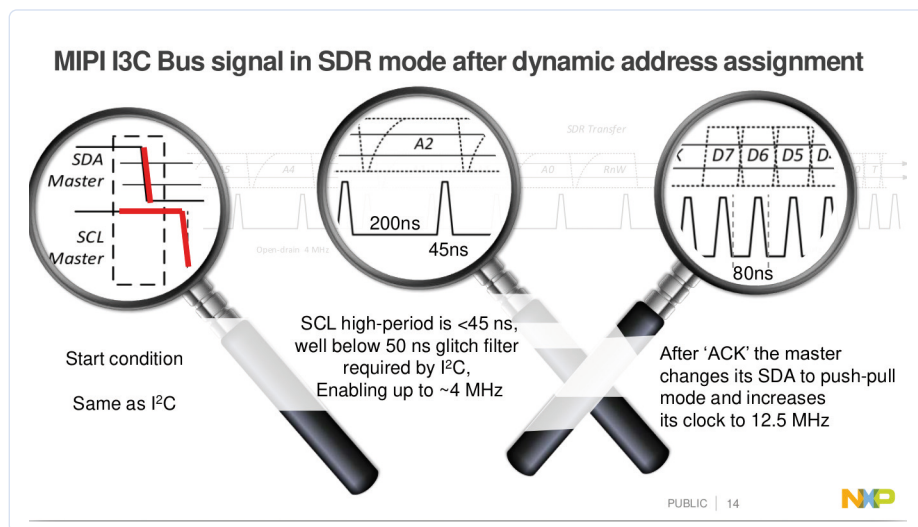
Deze vereenvoudiging gaat gepaard met een vereenvoudiging van de I/O-pinnen op de microcontroller of het klok-genererende IC. Voor I²C moeten aparte GPIO-pinnen worden gereserveerd in verband met het

eerder genoemde (50 ns) glitchfilter, maar de MIPI-specificatie specificeert alleen normale GPIO-pinnen die minimaal 4 mA kunnen leveren. Er staat ook expliciet in de specificatie dat glitchfilters niet vereist zijn.

De voordelen van snelheid

In **figuur 4** zie je enkele golfvormen op de I3C-bus. Het glitchfilter is bij I3C alleen van belang als het SCL-signaal op de bus een duty-cycle heeft die afwijkt van 50%, zoals bij I²C doorgaans het geval is. Bij I3C is de tijd dat het signaal hoog is gedefinieerd als korter dan 45 ns, dus een I²C-component die conform de standaard is geïmplementeerd kan helemaal niets met het signaal op de SCL-lijn.

Na ontvangst van zo'n 'onzichtbare' SCL-transactie, kan het klokgenererende IC zijn SDA-pin of SDA-lijn ook in push/pull-modus zetten en de werkfrequentie opvoeren tot



Figuur 4. I²C-componenten zien deze klokflank als 'laag' (bron: NXP-presentatie).

5 Pin Configuration and Functions

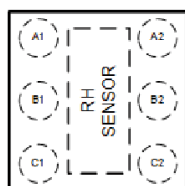


Figure 5-1. WLCSP (DSBGA) 6-Pin YPA Top View

Table 5-1. Pin Functions

PIN		I/O TYPE ⁽¹⁾	DESCRIPTION
NAME	NO.		
VDD	A1	P	Positive Supply Voltage
ADDR	B1	I	Address select pin – hardwired to VDD or GND. GND: slave address: 1000000 VDD: slave address: 1000001
GND	C1	G	Ground
SDA	A2	I/O	Serial data line for I ² C, open-drain; requires a pullup resistor to VDD
SCL	B2	I	Serial clock line for I ² C, open-drain; requires a pullup resistor to VDD
DRDY / INT	C2	O	Data ready/Interrupt. Push-pull output

(1) P=Power, G=Ground, I=Input, O=Output

Figuur 5. De HDC2010 heeft maar één adrespin, dus er kunnen er maximaal twee meedoen op een I²C-bus (bron: [9]).

12,5 MHz. In deze full-speed modus kan een overdrachtssnelheid tot 12,5 Mbps worden bereikt, conform de MIPI I3C-specificatie. Hogere overdrachtssnelheden kunnen met I3C worden bereikt in één van de drie HDR-modi (*High Data Rate*), maar die kunnen alleen worden gebruikt in twee bijzondere gevallen. Het eerste van die gevallen betreft een bussysteem dat *Pure I3C Bus* wordt genoemd. Dit bestaat uit uitsluitend

I3C-componenten. Het tweede betreft *Mixed Fast Bus* en kan I²C-componenten bevatten, maar alleen als die een glitchfilter hebben dat voldoet aan de specificatie.

De HDR-modi beginnen met een speciale reeks SCL-pulsen terwijl de SDA-lijn in de niet-actieve toestand verkeert. De eenvoudigste modus is HDR-DDR (*Double Data Rate*), waarmee snelheden tot 25 Mbps mogelijk zijn. Daarbij is elke flank van het SCL-

signaal een geldige trigger voor het ontvangen van data. Met de HDR-TSP-modus (*Ternary Symbol Pure*) zijn snelheden van meer dan 30 Mbps mogelijk. In deze modus worden de SCL- en SDA-lijn in combinatie gebruikt voor de data-overdracht. Nog een andere modus is HDR-TSL (*Ternary Symbol Legacy-inclusive*), die TSP beter laat omgaan met gemengde bussen.

Automatische adrestoewijzing

Een ander probleem van I²C is dat de adressen meestal fysiek moeten worden toegewezen aan de individuele componenten. Bij veel componenten (zoals de HDC2010 in **figuur 5**) houdt dat in dat de beschikbare adresruimte van de standaard niet volledig kan worden gebruikt.

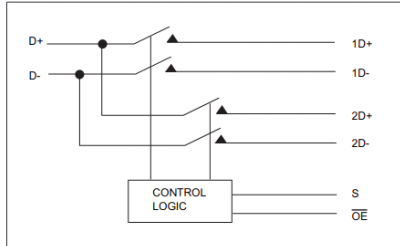
In plaats van de 10-bit adresmodus van I²C, beschikt I3C over dynamische zelfconfiguratie. Verder biedt *I3C Hot Join* de mogelijkheid om I3C-componenten op de bus aan te sluiten terwijl die actief is, net zoals dat kan bij USB. Op de achtergrond draait een *address arbitration*-procedure die voor het toewijzen van adressen aan de slave-IC's zorgt. Voor dat doel krijgt elk I3C-IC drie parameters: twee 8-bit velden met statusinformatie plus een unieke 48-bit UID. De specificatie gaat er van uit dat elke UID (die *Provisional ID* wordt genoemd) op de bus uniek is.

Bij het opstarten van de I3C-bus gaat het klokgenererende IC alle IC's op de bus nummeren en dynamisch een adres toewijzen. Dat lijkt op het arbitrageproces bij I²C: de I3C-component met de laagste UID-waarde krijgt als eerste een dynamisch adres. Het klokgenererende IC voert meerdere toewijzingscycli uit, totdat elk apparaat op de bus een tijdelijk ID heeft.

Adoptie van de standaard door fabrikanten

De lijst van MIPI-leden [5] staat vol bekende namen uit de wereld van de halfgeleiderfabrikanten, van GigaDevice tot ST en Solomon Systech. Vrijwel alle belangrijke bedrijven staan op de lijst. Maar als je in de praktijk op zoek gaat naar componenten, is er weinig te vinden. Zo zijn er bij Mouser momenteel alleen componenten zoals de PI3CSW12 van Dialog te vinden. Zoals je ziet in **figuur 6** zijn dat chips voor multiplexen en signaalconditionering. Renesas heeft, in elk geval voorlopig, een vergelijkbare benadering met chips zoals de

Block Diagram



Truth Table

S	OE	Function
X	H	Disconnect
L	L	D = 1D
H	L	D = 2D

Figuur 6. De momenteel verkrijgbare I3C-hardware beperkt zich tot voornamelijk signaalconditionering (bron: [10]).

IMX3102. In de aankondiging lezen we onder andere: "I3C biedt snelheden tot 12,5 MHz en verslaat daarmee oudere oplossingen zoals I²C met zijn snelheid van 1 MHz en analoge passieve snelle schakelaars."

Als je zoekt naar microcontrollers met I3C-ondersteuning, is de spoeling wel heel erg dun: de enige is momenteel de i.MX RT685 van NXP. Daar is ook een application note (AN12797) met voorbeeldsoftware voor. NXP biedt ook een soft core aan (soms gratis), en er is zelfs een speciale licentie voor gebruikers die geen MIPI-lid zijn [6]. Als je bereid bent om te betalen voor soft core intellectueel eigendom (IP), vind je bij leveranciers zoals Silvaco en Arasan Chip Systems kant-en-klare I3C-VHDL-modules.

Zoek je displays met I3C-ondersteuning (die vaak genoemd worden in de specificatie), dan moet je nog wachten. De enige fysieke

hardware die momenteel verkrijgbaar is, zijn enkele versnellingsopnemers (de BMI263 van Bosch en de LSM6DSO van ST) en de druksensor LPS22HH van Bosch.

Qua software is de situatie wel rooskleuriger: er is bij [7] een vrij gedetailleerde beschrijving beschikbaar van de interfacedriver in de Linux-kernel.

Hoe lang gaat het nog duren?

De grotere datasnelheid en de mogelijkheid om interrupts te genereren zonder speciale GPIO-ingangen kunnen mogelijk op den duur leiden tot een industriebrede adoptie van I3C, maar het is de vraag hoe lang dat nog gaat duren. Omdat er nog nauwelijks microcontrollers verkrijgbaar zijn met geschikte I3C-periferie, zou dat volgens mij nog wel eens tegen kunnen vallen. 

210640-03

Waarom niet Master en Slave?

De auteur, doorgaans aartsconservatief in politieke zaken, geeft de voorkeur aan de aanduidingen 'klokgenererende component' en 'klokvolgende component'. Zijn ervaring bij het onderwijzen van generaties van studenten heeft aangetoond dat die manier van denken tot meer begrip leidt (en bovendien politiek correct is).

Hoeveel busdeelnemers?

I3C is restrictiever dan I²C met betrekking tot het toelaatbare aantal IC's op de bus, waarschijnlijk vanwege de hogere snelheid en de geringere capaciteit van de pinnen om stroom te leveren of op te nemen. Bij eerdere versies van de standaard zei MIPI dat er maximaal elf apparaten per bus zouden worden ondersteund. Maar die beperking, die door middel van een simulatie van de elektrische eigenschappen was bepaald, staat niet meer in de standaard. Van de theoretisch mogelijke 2⁷ (128) dynamische adressen zijn er ongeveer 90 beschikbaar, afhankelijk van de busconfiguratie.

Vragen of opmerkingen?

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



GERELATEERDE PRODUCTEN

- **V. Himpe, Mastering the I²C Bus (Elektor 2011) (SKU 15385)**
www.elektor.nl/15385
- **V. Himpe, Mastering the I²C Bus (Elektor 2011) (PDF, SKU 16193)**
www.elektor.nl/16193

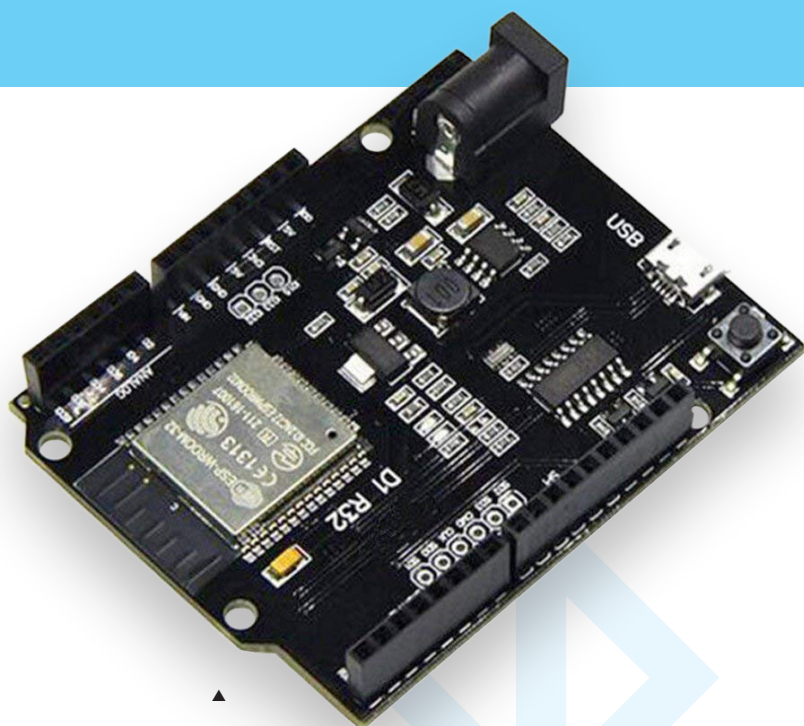
WEBLINKS

- [1] Discussies over I²C: http://www.verycomputer.com/31_b0b5a85949fcce82_1.htm
- [2] MIPI I3C-webpagina: www.mipi.org/specifications/i3c-sensor-specification
- [3] MIPI-lidmaatschap is kostbaar: www.mipi.org/join-mipi
- [4] I3C-basispecificatie: <https://resources.mipi.org/blog/mipi-alliance-delivers-new-i3c-basic-specification>
- [5] List van MIPI-leden: www.mipi.org/devcon/membership/all-member-directory
- [6] NXP soft core-IP: <https://bit.ly/3qWAvD3>
- [7] API in de Linux-kernel: www.kernel.org/doc/html/latest/driver-api/i3c/protocol.html
- [8] MIPI I3C-white paper: <http://bit.ly/2gld6BL>
- [9] Datasheet van de HDC2010 van Texas Instruments: <http://www.ti.com/lit/gpn/hdc2010>
- [10] Datasheet van de PI3CSW12 van Diodes Inc: <http://www.diodes.com/assets/Datasheets/PI3CSW12.pdf>

BlueRC:

IR-afstandsbediening met smartphone en ESP32

universeel en aanpasbaar



Figuur 1. WeMos D1 R32 ESP32 WiFi/Bluetooth-ontwikkelboard.

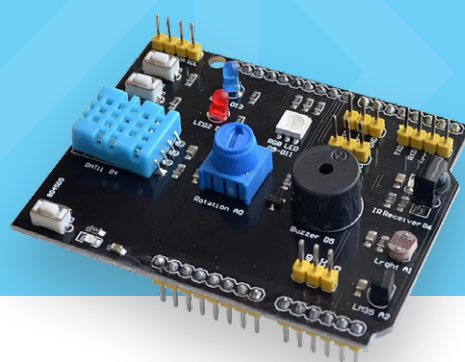
Xin Wang (Duitsland)

Ooit waren IR-zenders op mobiele telefoons heel gebruikelijk, maar dat is nu niet meer het geval. Dit project biedt een oplossing: gebruik je mobiele telefoon met zijn Bluetooth-connectiviteit om een ESP32-board aan te sturen dat zowel bekende infraroodcommando's kan versturen als nieuwe kan leren.

Ik wilde mijn IR-gestuurde apparaten ook kunnen bedienen met mijn telefoon. Daarom deed ik wat onderzoek en vergeleek verschillende projecten op het web. De meeste daarvan werkten met voorgedefinieerde codes, wat betekent dat de betreffende IR-codes vooraf bekend moesten zijn en in een matrix moesten worden opgeslagen. Ik wilde een zelflerende afstandsbediening bouwen waarvoor geen voorkennis van de codes van bestaande afstandsbedieningen nodig was. De codes moesten in real time en 'tijdens bedrijf' door de schakeling worden bijgewerkt.



Figuur 2. Gebruikersinterface van de BlueRC V1.1 Android-app.



Figuur 3. Het Keyestudio Easy Module Shield V1 voor Arduino (bron: flyrobo.in).

De hardware

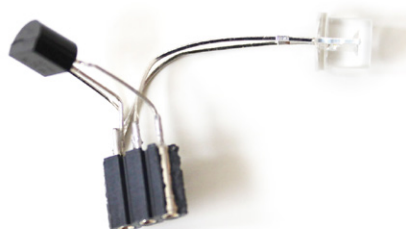
Er zijn slechts twee belangrijke bouwstenen voor dit project:

- Een microcontroller-board (**figuur 1**) met een aangesloten infrarood-ontvanger voor het aanleren van bestaande codes van andere afstandsbedieningen, en een infrarood-zender voor de hoofdtak: het verzenden van commando's naar de TV, set-top box enzovoort. Het board slaat ook de geleerde codes op.
- Een Android-telefoon om een app te draaien die enkel fungeert als de gebruikersinterface voor het microcontroller-board. De app voorziet in een afstandsbediening op het display (**figuur 2**) waarop je de toetsen kunt indrukken die nodig zijn om je IR-gestuurde apparaten te bedienen. Je kunt de leerprocedure starten met een toetsdruk in deze gebruikersinterface, maar die leerprocedure zelf wordt uitgevoerd op de microcontroller-print.

Ik heb Bluetooth gekozen voor de communicatie tussen de telefoon en de IR-zenderhardware, vanwege het lage vermogen en de geringe latentie, vergeleken met WiFi. Een commercieel verkrijgbare oplossing die aan alle hardwarevoorwaarden voldeed bleek helaas onvindbaar: niets was 100% geschikt. Voor de microcontroller koos ik daarom voor een ESP32-board, omdat dat zowel Bluetooth als WiFi aan boord heeft. Het D1 R32 ontwikkel-board van WeMos heeft een Arduino UNO-compatibele pin-layout en werkt goed met de Arduino IDE.

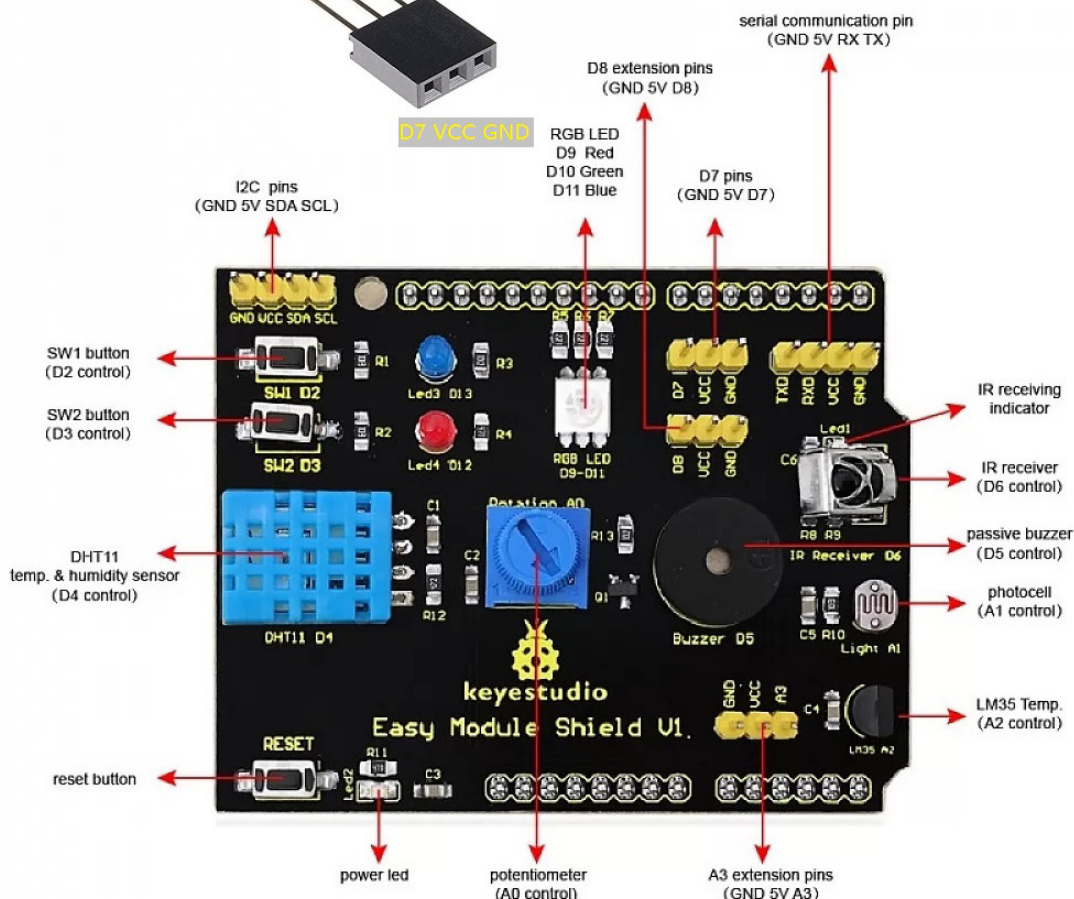
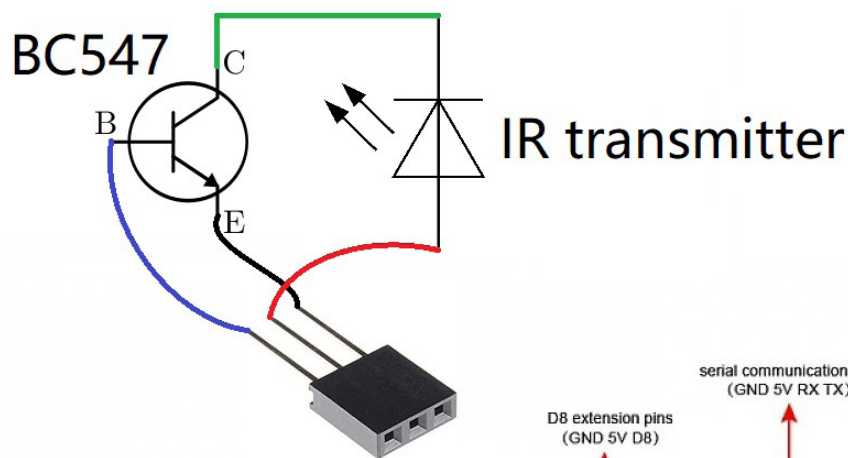
Ik voegde er een shield met meerdere sensoren (**figuur 3**) aan toe. Dit shield heeft talloze zaken aan boord, zoals een vochtigheidssensor, een zoemer, een potentiometer, een fotocel, een RGB-LED en een paar knoppen – een heleboel veelzijdigheid ten behoeve van toekomstige domotica-projecten. Ik heb de sensoren gebruikt om de temperatuur en vochtigheid in de Android-app weer te geven, maar de belangrijkste sensor op het shield is natuurlijk de infrarood-ontvanger!

Hoewel het shield een ingebouwde IR-ontvanger heeft, is er geen zender, dus moest ik zelf een driver/zender in elkaar knutselen. Deze bestaat uit slechts 3 componenten – een 3-polige busstrip, een BC547 NPN-transistor en een IR-LED, gemonteerd als in **figuur 4**. Na het solderen prikte ik de busstrip op poort-pin D7 op het sensor-shield, dat poort D14 gebruikt als uitgang om mijn IR-zender aan te



Figuur 4. Mijn zelfbouw IR driver/zender-module.

Figuur 5. Het shield en de aansluiting van de IR driver/zender-module (bron: flyrobo.in).



sturen, terwijl de IR-ontvanger van het schild standaard poort D27 gebruikt als ingang.

In het 'schema' van **figuur 5** kun je zien hoe ik een en ander heb aangesloten.

De software

De software is relatief eenvoudig: de ESP32 fungeert als Bluetooth-server, en een Android-app doet dienst als client. De ESP32 wacht gewoon op Bluetooth-berichten van de Android-app. De Bluetooth-communicatie tussen de twee apparaten is minimaal, omdat toetsaanslagen van de afstandsbediening – en het commando om de leercurve te beginnen ("^") – door de Android-app als afzonderlijke tekens naar de ESP32 worden gestuurd. Eerst probeerde ik de standaard IR-bibliotheek voor Arduino. Die werkte prima, maar wordt niet volledig

ondersteund op de ESP32, omdat deze gebruik maakt van een eigen Remote Control Transceiver (RMT) in de vorm van onboard-hardware, met veel protocol en modulatieloga ingebouwd voor IR-signalen. Na veel vallen en opstaan vond ik eindelijk een goed werkende IR-bibliotheek voor de ESP32 (*IRremoteESP8266*) van Ken Shirriff et al [1]. Bedankt, Ken.

Oorspronkelijk wilde ik de IR-codes opslaan in EEPROM, maar toen ik de firmware schreef, realiseerde ik me dat de ESP32 helemaal geen EEPROM heeft. Het heeft echter wel de mogelijkheid om gegevens permanent op te slaan in het flash-geheugen via de *Preferences.h*-bibliotheek [5]. Omdat mijn originele afstandsbediening een RC5-exemplaar was, heb ik alleen de RC5-codering geïmplementeerd. Eventueel moet je de Arduino-sketch aanpassen als je eigen afstandsbediening een ander protocol gebruikt.



Programmering

Voor het WeMos-board kun je de BlueRC Arduino-sketch van [2] downloaden. Het is een enkel .ino-bestand dat je kunt uploaden met de Arduino IDE.

Volg deze stappen om de Android-app te installeren:

1. Download op je Android-toestel **BlueRC-app-V1.1.apk** van de GitHub pagina op [3]. Je moet je smartphone zo instellen dat het downloaden van apps van externe bronnen (anders dan Google Play Store) is toegestaan.
2. Schakel het WeMos-board met de **BlueRC.ino**-firmware in.
3. Start de Android-app. Als Bluetooth niet actief is, vraagt de app om toestemming; tik op **Accept**.
4. Tik op het "kebab menu" rechtsboven en vervolgens op **Connect BT Device**. Als dit de eerste keer is dat je een koppeling met de ESP32 maakt, tik dan op **Scan for Devices** en wacht tot het ESP32BlueRC-Shield verschijnt onder **Other Available Devices**. Tik hierop en bevestig de koppeling.
5. In de statusbalk bovenaan moet nu **connected:ESP-32BlueRC-Shield** staan.

Zodra je de hardware hebt gekoppeld, hoeft je het scanproces niet opnieuw uit te voeren – tik voortaan gewoon op **ESP32BlueRC-Shield** onder **Paired Devices**. Nu de app is geïnstalleerd en werkt, kun je de codes gaan aanleren:

1. Tik lang op de LEARN-knop in de UI; de app zal reageren met "Learning begins, please press a button within 10 seconds..." De Android-app stuurt dan het speciale teken "A" naar de ESP32, waarna de blauwe LED op het schild oplicht.
2. Tik in de gebruikersinterface op de knop waaronder het ESP32-board de code moet aanleren. De smartphone zal het corresponderende teken naar de ESP32 sturen, waarna deze op de volgende stap wacht.
3. Richt je bestaande IR-afstandsbediening op het ESP32-board en druk op de betreffende knop. De IR-code voor deze knop wordt nu automatisch opgeslagen in het geheugen van de ESP32.

4. Herhaal stappen 1...3 voor alle knoppen die je wilt aanleren.
5. Sluit de app door op **Menu Exit Program** te tikken.

Klaar. Je kunt nu je IR-bestuurde apparaten bedienen vanaf je BlueRC-apparaat via de app op je Android-telefoon.

Om problemen met de ESP32 op te lossen, kun je hem het beste uit- en weer aanzetten!

Een video van het apparaat in actie is te zien op mijn YouTube-kanaal [4]. 


Bekijk dit project op video!
<https://youtu.be/Kb-TdF84Oz4>



220103-03

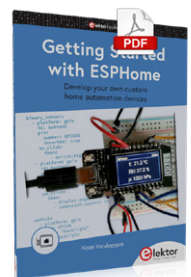
Vragen of opmerkingen?

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



Gerelateerde producten

- > **Dogan and Ahmet Ibrahim, The Official ESP32 Book (PDF, SKU 18330)**
www.elektor.nl/18330
- > **Koen Vervloessem, Getting Started with ESPHome (PDF, SKU 19739)**
www.elektor.nl/19739

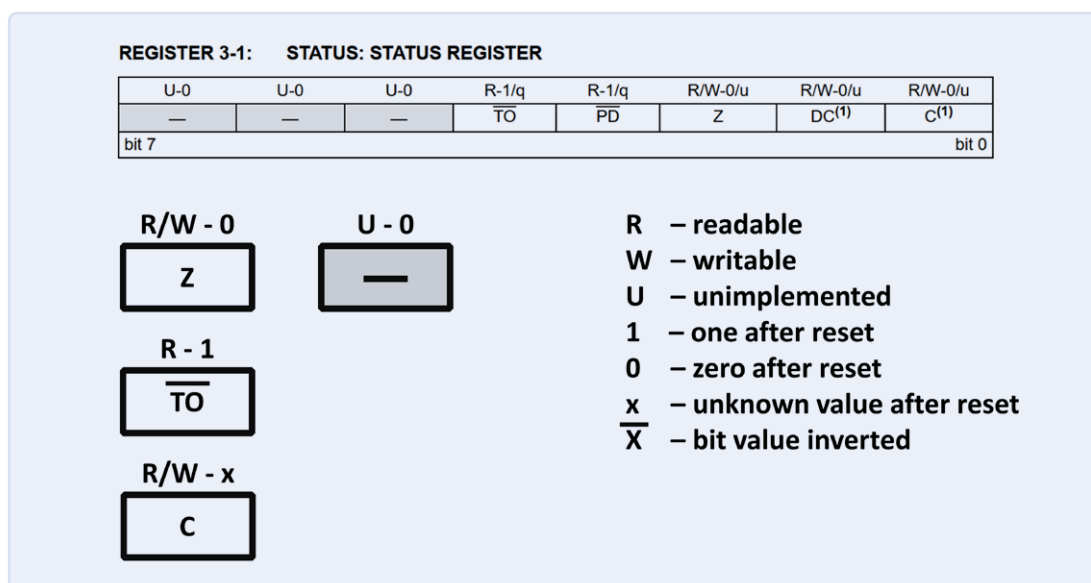


WEBLINKS

- [1] IRremoteESP8266-bibliotheek: <https://www.arduino.cc/reference/en/libraries/irremotesp8266/>
- [2] BlueRC Arduino-sketch voor ESP32: <https://elektor.link/BlueRCESP32>
- [3] BlueRC Android-app: <https://elektor.link/BlueRCApp>
- [4] Het BlueRC-systeem in actie: <https://youtu.be/Kb-TdF84Oz4>
- [5] ESP32 Save Data Permanently using Preferences Library: <https://randomnerdtutorials.com/esp32-save-data-permanently-preferences/>

Microcontroller-documentatie verklaard

deel 2: registers en blokschema's



Figuur 1. De beschrijving van de bits van een register kan er ingewikkeld uitzien (bron: Microchip Technology).

Stuart Cording (Elektor)

Of we het nu leuk vinden of niet, we zullen met microcontroller-documentatie moeten werken. In het eerste deel van deze artikelreeks [1] hebben we gezien welke informatie er in de datasheet van een microcontroller te vinden is. In deze aflevering gaan we een stuk dieper en kijken we hoe gebruikers worden geïnformeerd over de functionaliteit van de chip, zoals de registers en de blokschema's.

De eerste pagina's van de datasheet

Nu we weten wat er zoal in een datasheet te vinden is, kunnen we naar de details gaan kijken. Datasheets beschrijven de zaken op een wat eigenaardige manier, en elke fabrikant doet dat weer op een eigen manier. We blijven ons concentreren op de 8-bits PIC16F18877-microcontroller van Microchip Technology [2], dus het is van belang om die datasheet te downloaden zodat je dit verhaal kunt volgen.

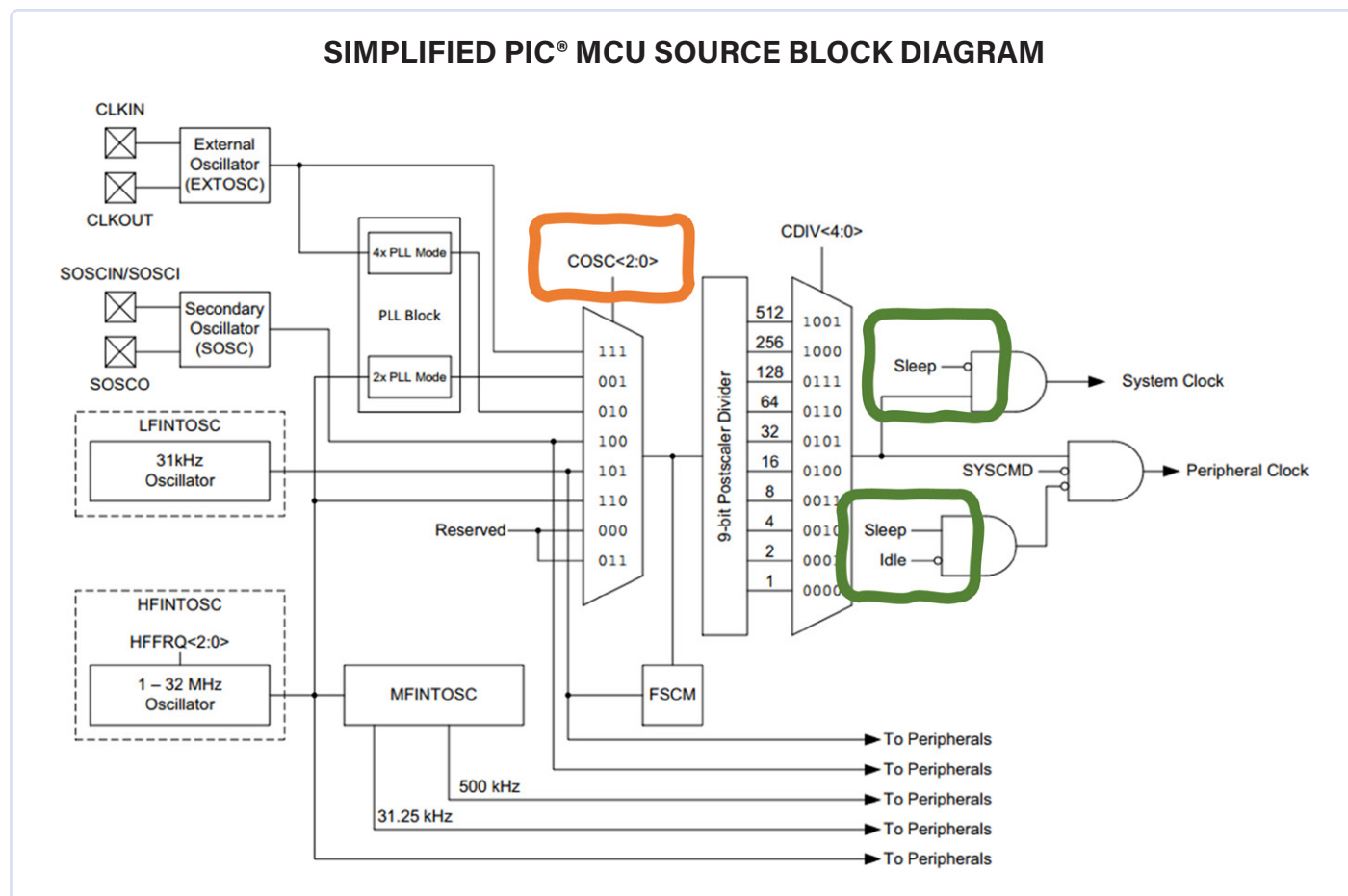
Na het downloaden en openen van de datasheet word je geconfronteerd met de eerste twee pagina's, die een korte beschrijving geven van de microcontroller, plus een overzicht van de belangrijkste eigenschappen en de periferie. De *Core Features* vormen geen onafhankelijke periferie; het gaat om functies die nauw verbonden zijn met de processorkern. Het gaat om functies zoals de brown-out schakeling die detecteert wanneer de voedingsspanning onder het vereiste niveau daalt, en watchdog-timers, die de microcontroller resetten en vaak worden gebruikt in veiligheidskritische systemen.

De derde pagina komt al meer ter zake en vertelt hoeveel en welke periferie we krijgen, en hoeveel RAM, flashgeheugen en EEPROM aanwezig is. De pagina's vier tot en met zeven geven basisinformatie over de behuizingen en de pinning van de diverse uitvoeringen. De *Pin Allocation Tables* zijn een essentieel onderdeel van elke microcontroller-datasheet. Met zoveel mogelijkheden op elke chip hebben microcontrollers tegenwoordig meer functionaliteit dan aansluitpinnen. Daarom kunnen verschillende mogelijkheden (en soms zelfs meer dan één) worden toegewezen aan een enkele pin. Bij deze controller zien we dat pin RA2 kan worden gebruikt als een ADC-ingang, een analoge spanningsreferentie, een comparator-ingang, een DAC-uitgang of als een interrupt-pin. Soms kan een bepaalde functie worden toegewezen aan verschillende pinnen, wat meer flexibiliteit biedt. De toewijzing van functionaliteit aan pinnen kan dus bijzonder ingewikkeld zijn.

Registerbeschrijvingen begrijpen

Registerbeschrijvingen bestaan uit twee delen: de naam van het register en de namen van de bits (of bitgroepen) in het register.

Pagina 38 van deze datasheet beschrijft de functionaliteit van het STATUS-register in de processor. Dit register bestaat uit bits die zijn geïmplementeerd en bits die niet zijn geïmplementeerd. De geïmplementeerde bits zijn gemarkeerd met een combinatie van R en/of W die aangeeft of we ze kunnen lezen en/of schrijven. Na het streepje komt meestal een cijfer: "0" of "1". Dit geeft de waarde van dit bit aan na een reset. Bij sommige bits staat een x, wat aangeeft dat de waarde na een reset onbekend is. De waarde van sommige bits is afhankelijk van welke gebeurtenis de microcontroller uit zijn reset-toestand heeft gehaald. Als de microcontroller gewoon is ingeschakeld, is hij gewekt door een power-on reset (POR). Als de voeding een brown-out reset (BOR) heeft veroorzaakt, kan er een andere waarde in een specifiek bit staan. In dit voorbeeld zijn zulke "reset-afhankelijke" bits gemarkeerd met een q. Niet-geïmplementeerde bits kunnen soms gevaarlijk zijn. Volgens deze datasheet zouden de hoogste drie een "0" moeten bevatten. Maar het is niet duidelijk wat er gebeurt als je naar deze bits probeert te schrijven. De datasheet zou een algemene aanbeveling kunnen geven. Als die er niet is, is het waarschijnlijk verstandig om ervoor



Figuur 2. Blokschema van de klokperiferie, waarin de bron van de kloksignalen kan worden geselecteerd en de kloksignalen kunnen worden in- en uitgeschakeld (bron: Microchip Technology).



Niet-geïmplementeerde bits nul maken

```
myValue &= 0b00011111;    // clear 3 most-significant bits
                          // (unimplemented bits)
STATUS = myValue;         // write 'myValue' into STATUS register
```

te zorgen dat wanneer je iets naar die registers schrijft, de niet-geïmplementeerde bit nul worden gemaakt. In C/C++ kan dat als volgt:

```
myValue &= 0b00011111; // clear 3 most-significant bits
                      // = (unimplemented bits)
STATUS = myValue;      // write 'myValue' into
                      // STATUS register
```

Opmerking: het is wel aan te raden om duidelijk commentaar bij deze code te zetten, om te voorkomen dat in de toekomst iemand je veiligheidsmaatregel “wegoptimaliseert”.

Een streepje boven de naam van een bit geeft aan dat de waarde geïnverteerd is. Dat geldt bijvoorbeeld voor het bit TO (“Time Out”). Als een time-out optreedt, wordt de waarde dus niet “1”, zoals je zou verwachten, maar “0”. Voor de programmeur is het vaak onduidelijk waarom zulke bits geïnverteerd zijn; je moet ermee leren leven en de bits in je code correct behandelen.

Soms is een groep bits nodig om iets te configureren, zoals de prescaler voor een klokbron, of de baudrate voor een seriële interface. Zulke bitgroepen worden aangeduid als `BIT_GROUP<X:Y>`, waarbij `X` het meest significante bit van de groep is en `Y` het minst significante bit. En opgepast: soms zijn de bits die bij een groep horen geen opeenvolgende bits in het register, soms zijn ze zelfs verdeeld over meer dan één register!

Tijd voor de klok

De klok-periferie is het belangrijkste deel van de microcontroller. Hier wordt bepaald wat er wordt gebruikt als kloksignaal voor de processor en voor de periferie in de chip. Het is nuttig om dit

grondig te bestuderen. Je vindt dit op pagina 110, Normaal gesproken wordt deze periferie eenmalig geconfigureerd bij het starten van het programma en daarna niet meer veranderd. Als de configuratie wordt veranderd, kan dat gevolgen hebben voor alle periferie, van de bemonsteringsfrequentie van de ADC tot de baudrate van de seriële UART en de CAN-interface.

In dit voorbeeld hebben we enkele externe bronnen (kristaloscillatoren of keramische resonatoren) en een paar interne bronnen. Die laatste zijn misschien niet nauwkeurig genoeg (vooral bij temperatuurveranderingen) als betrouwbare klok voor sommige interfaces, zoals de UART. Waarschijnlijk wordt dat in de datasheet vermeld. Anders moet je de nauwkeurigheid checken in het hoofdstuk *Electrical Characteristics*.

In het schema zie je dat een groep van drie bits, `COSC<2:0>`, in een register dat te maken heeft met de oscillator, wordt gebruikt om te kiezen tussen de beschikbare klokbronnen met behulp van een multiplexer. Om één of andere reden wordt alleen de groep bits genoemd in zulke schema's en niet het register waar ze bij horen. De beste manier om te vinden in welk register die groep zit, is ernaar te zoeken in het PDF-bestand. Ze kunnen zomaar verstopt zitten in een register van ongerelateerde periferie! Multiplexers worden vaak gebruikt in blokschema's waar tussen signalen kan worden geschakeld.

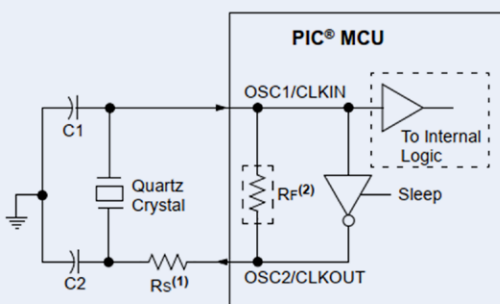
De uitvoer van dit blok levert de *System Clock*, waarschijnlijk voor de processor, en de *Peripheral Clock* voor de periferie. Naar die begrippen kan worden verwezen in andere delen van de datasheet, vooral bij de bespreking van de *Low Power*- of *Sleep*-modi. Uit de groen gemarkeerde gedeelten kunnen we al concluderen dat er een *Idle*-modus is die alleen de klok voor de periferie uitschakelt, en een *Sleep*-modus die de klok voor zowel de periferie als voor de processor uitschakelt.

Als een kristal of keramische resonator moet worden gebruikt, worden er waarschijnlijk aanwijzingen gegeven over hoe die moeten worden aangesloten en over de beste print-layout. Dat is ook hier het geval (pagina 112): er zijn twee condensatoren nodig en eventueel een seriële weerstand. Het feit dat dit blokschema vergezeld gaat van vier application notes over het ontwerpen van oscillatoren, geeft al aan dat dit nog niet zo eenvoudig is.

Voorbij de registers en blokschema's

We hebben nu de basis besproken van hoe registers worden beschreven en hoe we het blokschema van de klokperiferie moeten interpreteren. We hebben zelfs een blik geworpen op één van de oscillatoren en het voorbeeldschema voor het gebruik van een kristal. In het volgende deel van deze serie gaan we kijken naar een ander belangrijk periferie-element: het resetblok. En we gaan ook op zoek naar alles wat het datasheet ons niet vertelt.

QUARTZ CRYSTAL OPERATION (LP, XT, OR HS MODE)



Figuur 3. Aanbevolen schema van een kristaloscillator (bron: Microchip Technology).

Als je belangstelling hebt voor informatie die je instap in de wereld van de microcontrollers gemakkelijker kan maken, werp dan eens een blik op onze inleidende artikelen [3] en boeken. [K](#)

vertaling: Evelien Snel — 230101-03

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een email naar de auteur via stuart.cording@elektor.com.



Gerelateerde producten

- > T. Hanna, *Microcontroller Basics with PIC*, Elektor 2020 (SKU 19188)
www.elektor.nl/19188
- > A. Pratt, *Programming the Finite State Machine*, Elektor 2020 (SKU 19327)
www.elektor.nl/19327

WEBLINKS

- [1] Stuart Cording, "Microcontroller-documentation verklaard (deel 1)": www.elektormagazine.nl/magazine/elektor-295/61525
- [2] 8-bit PIC16F18877-microcontroller: <https://microchip.com/wwwproducts/en/PIC16F18877>
- [3] Tam Hanna, "PIC's programmeren: van de grond af": <https://www.elektormagazine.nl/magazine/elektor-157/59016>

MagPi, het officiële Raspberry Pi-magazine



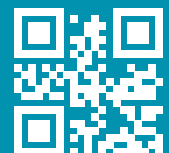
Zes edities
van MagPi
Magazine



Toegang tot
het online
MagPi-archief

12 maanden
100+ projecten
1 prijs
€ 54,95

NU TE BESTELLEN OP
WWW.MAGPI.NL/ABO



MagPi 
www.magpi.nl Magazine

Automatisering van **testen** en **meten**

testapparatuur programmeren om je te dienen

Stuart Cording (Elektor)

Testen kan eentonig en steeds weer hetzelfde zijn, en het opsporen van fouten is een buitengewone uitdaging. Waarom dan geen gebruik maken van de communicatie-interfaces op test- en meetapparatuur om het gemakkelijker te maken? We verkennen diverse apparaten die op eenvoudige wijze met Python bestuurd kunnen worden, zonder de noodzaak van dure software of licenties.

Voor de meeste ontwikkelaars staat test- en meetapparatuur op de werkbank te wachten om een taak uit te voeren. Maar draai de meeste van die apparaten eens om, en je vindt vaak een communicatie-interface aan de achterkant. In combinatie met de juiste software krijg je de mogelijkheid om metingen te besturen en resultaten te verzamelen voor latere analyse. Dit kan bijzonder nuttig zijn bij het zoeken naar sporadische gebeurtenissen en storingen of het testen van een toepassing met verschillende systeemparameters. Dit is ook de manier waarop het testen aan het einde van de productielijn en het sorteren van componenten aan de hand van prestaties kan worden geautomatiseerd.

Afstandsbediening van labvoedingen

Een goed uitgangspunt is de voeding. Voor de meeste toepassingen is het nodig om de voeding te herstarten om een harde reset uit te voeren. Bij meer geavanceerde tests zal de applicatie in kwestie tot aan de uiterste grenzen van de toegestaneingangsspecificatie werken, en kan het nodig zijn om overspan-

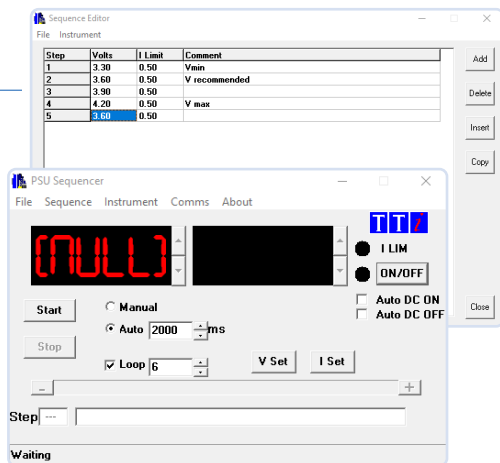
ningstests uit te voeren. Dit is gebruikelijk in de auto-industrie, waar pulsen tot 87 V moeten worden verdragen gedurende 400 ms (ISO 7637-2). Een andere veel voorkomende storing in samenhang met de voedingsspanning is een langzaam stijgende of dalende spanning. Onder dergelijke omstandigheden komen schakelingen vaak in een brown-out toestand terecht waarvan ze zich niet meer kunnen herstellen. Ten slotte helpt het voor geautomatiseerde testsystemen om apparatuur te hebben die correct kan worden ingesteld voor zowel spanning als maximale stroom, vooral wanneer dezelfde opstelling wordt gebruikt voor verschillende producten.

Laboratoriumvoedingen zijn de laatste jaren sterk in prijs gedaald. Om de toestroom van goedkope apparatuur te compenseren, breiden de bekendere merken meestal het scala aan mogelijkheden uit, en zijn niet bereid of in staat om alleen op prijs te concurreren. Het is onbekend of dit het geval is bij Aim and Thurlby Thandar Instruments (Aim-TTi) in Cambridge (Groot-Brittannië), maar ze maken hun belofte van 'meetbae meerwaarde' wel waar. Ze bieden een uitgebreide serie



Figuur 1. Twee modellen uit de EL-R serie gelijkspanningsvoedingen bieden een seriële interface (RS-232/virtuele COM via USB) ter ondersteuning van testautomatisering (bron: Aim-TTi).

gelijkstroomvoedingen, en hun instapmodel uit de EL-R serie (**figuur 1**) is een nadere kennis-making waard, vooral als je speelt met het idee van je eerste geautomatiseerde testopstelling. Deze voedingen gebruiken een ruisarme lineaire regeling voor enkele, dubbele en drievoudige uitgangen. Ze hebben geen ventilator maar vertrouwen op convectiekoeling, en variëren in vermogen van 30 tot 130 W. Met één of twee rode LED-displays en analoge regeling; sommige modellen zijn ook uitgerust met remote sense-terminals. Twee interessante modellen zijn de EL302P met één uitgang, met RS-232, en de EL302P-USB met USB, beide 60 W bij 0...30 V en 0...2 A. De apparaten worden geleverd met software-drivers voor de interface. Daarnaast is op de



Figuur 2. De PSU Sequencer-software van Aim-TTi biedt eenvoudige, repeterende cycli van spanning- en stroominstellingen (bron: Aim-TTi).

website van de leverancier het hulpprogramma 'PSU Sequencer' beschikbaar [1], waarmee door de gebruiker geconfigureerde spanning/stroom-instellingen handmatig of automatisch kunnen worden doorlopen (figuur 2). Bovendien kunnen vooraf voorbereide meetreeksen uit een spreadsheet worden geïmporteerd.

Python voor voedingsregeling

Het ontwikkelen van je eigen besturingssoftware, afgestemd op jouw testbehoeften, is ook eenvoudig. De hardware RS-232 interface werkt van 600 tot 9.600 baud, en de USB-interface verschijnt als een virtuele COM-poort. De commando's staan allemaal in de handleiding [2], en met een beetje planning is het eenvoudig genoeg om een bibliotheek te ontwikkelen die alle commando's omzet in duidelijk benoemde functies of methoden. Een optie is om een Arduino en een RS-232 transceiver te gebruiken om de voeding te regelen. Dit heeft het voordeel dat ook andere testfuncties kunnen worden geïntegreerd, zoals het schakelen van signalen met behulp van relais of het inlezen van analoge en digitale signalen. Als alternatief biedt Python de module pySerial [3]. Met wat eenvoudige codering (listing 1) kan de commando-interface worden geïmplementeerd als een Python-module en kan geautomatiseerde besturing worden gerealiseerd. Met wat leeswerk in de documentatie is het ook mogelijk om een TCP/IP-naar-serieel verbinding te implementeren [4]. Deze functionaliteit is gedefinieerd in de experimentele RFC2217 [5] memo en staat een PC op afstand toe om een seriële interface te configureren en communicatie te implementeren.

Signalen onderzoeken

Oscilloscopen kunnen ook op afstand bediend worden. Met hun uitgebreide mogelijkheden, van opname van analoge en digitale signalen

tot FFT's, kunnen ze gebruikt worden voor diverse geautomatiseerde tests. Sommige storings zijn bijvoorbeeld moeilijk te vinden door een complexe reeks gebeurtenissen die in een specifieke volgorde moeten plaatsvinden om ze te veroorzaken. Als je team eenmaal heeft bepaald hoe de storing wordt veroorzaakt, is de volgende fase het configureren van de oscilloscoop om relevante signalen op te vangen die helpen de primaire oorzaak vast te stellen.

Boards als Arduino en Raspberry Pi zijn geweldig voor het snel ontwikkelen van een testopstelling die een storing herhaaldelijk kan opwekken, met behulp van analoge uitgangen en digitale signalen, en indien nodig aangevuld met een relais of MOSFET. Ze kunnen ook nauwkeurige triggersignalen leveren voor een oscilloscoop, zodat het juiste deel van de relevante signalen wordt opgeslagen voor latere analyse.



Listing 1. Eenvoudig Python-script om een PSU ID-string te verkrijgen met pySerial.

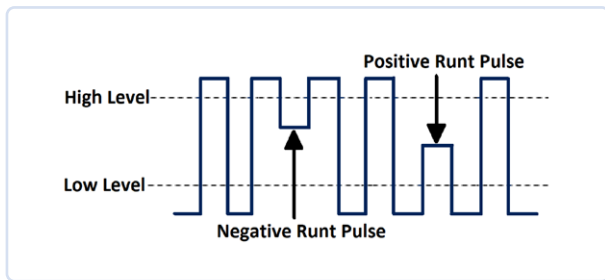
```
import serial
ser = serial.Serial('/dev/ttyUSB0', 9600, timeout=0)

# open serial port
# confirm port was really used
print(ser.name)
ser.write(b'*IDN?')
# request PSU's ID strings
response = ser.readline()
# collect response
print(response)
# output PSU ID string
ser.close()
# close port
```

Veel oscilloscopen hebben USB- en LAN-interfaces, maar sommige bieden alleen ondersteuning voor eigen software of toegang tot een webserver voor configuratie. Er is echter een USB-klasespecificatie voor testen en meten beschikbaar, die bekend staat als USBTMC [6]. Net als opslagapparatuur zoals USB-drives, en human-interface devices (HUD) zoals je muis en toetsenbord, biedt deze klasse voorgedefinieerde commando's die geschikt zijn voor de interfacing met test- en meetapparatuur. Apparaten zoals de B&K Precision 2567B [7], een 200-MHz, 2-GSa/s mixed signal oscilloscoop (MSO), ondersteunen deze interface. Met vier analoge kanalen, een 16-kanaals digitale poort, een ingebouwde 50-MHz arbitraire golfvormgenerator en geavanceerde triggers is hij eenvoudig te configureren met behulp van zijn grote 10,1" capacitieve touchscreen (figuur 3). Maar het gaat net zo gemakkelijk via USB.



Figuur 3. Met een USBTCM-compatibele USB-interface is de B&K Precision 2560B-serie MSO's gemakkelijk te automatiseren met Python (bron: B&K Precision).



Figuur 4. De runt-trigger is een slim mechanisme om pulsen te vinden die niet volledig stijgen of dalen (bron: B&K Precision).



Figuur 5. De schijfvormige Moku:Lab is een twaalf-in-één instrument, en kan geconfigureerd worden met behulp van een Python API (bron: Liquid Instruments).

Configuratie op afstand

Ook hier is Python de aangewezen programmeertaal, dankzij het USBTMC-project [8] van Alex Forencich, dat gehost wordt op GitHub [9]. De module draait onder Linux, maar kan vragen dat de machtigingen op de juiste manier worden aangepast. Onder Windows moeten eerst *PyUSB* [10] en *libusb* [11] worden geïnstalleerd.

De eerste stappen zijn relatief eenvoudig. Alvorens te beginnen zijn de vendor- (VID) en product-ID (PID) nodig, twee USB-specifieke referentiewaarden die een USB-product identificeren. Onder Linux gaat dat eenvoudig via de opdrachtregel met `lsusb` zodra het apparaat is aangesloten. De resulterende lijst geeft de benodigde details. In Windows kunnen deze waarden worden bepaald via Apparaatbeheer en de eigenschappen van het apparaat. In beide gevallen zijn de resulterende waarden 16-bit hexadecimaal.

Alles wat dan nog gedaan moet worden is het importeren van de *usbtmc*-module in je Python-code en het gebruiken van de beschikbare programmeerinterface (API). Het lijkt veel op het printen van tekst naar het scherm en het evalueren van toetsenbord invoer. USBTMC is eigenlijk gewoon een wrapper om te communiceren met geschikte test- en meetapparatuur. De besturingscommando's zijn ASCII-strings

met diverse opties, zoals beschreven in de programmeerhandleiding van de oscilloscoop [12]. Indien beschikbaar kunnen ook VISA resource-strings [13] gebruikt worden.

De VID en PID zijn uniek voor het product, niet voor het individuele apparaat. Als er dus twee of meer van hetzelfde apparaat zijn aangesloten, kunnen ze onafhankelijk van elkaar worden aangesproken met een derde parameter, het serienummer. Dit staat meestal op een etiket of kan worden verkregen met het `*IDN?` commando (listing 2).

Tegenwoordig onderscheiden oscilloscopen zich niet veel meer van elkaar afgezien in bandbreedte, samplingsnelheid en geheugendiepte. Maar af en toe duikt er een kleine voorziening op die interessant is. De B&K-serie heeft een runt-trigger (figuur 4), een mogelijkheid die een triggerpuls levert wanneer een signaal de ene drempelwaarde (bijvoorbeeld hoog) passeert, maar de andere (bijvoorbeeld laag) niet. Bij storingsanalyse kan elk detail behulpzaam zijn.

Testapparatuur zonder display

Red Pitaya doorbrak de standaard op het gebied van testapparatuur en vroeg zich af of een meetapparaat wel een eigen display nodig heeft. Dankzij een krachtige FPGA en een Ethernet-interface levert je PC of laptop de

gebruikersinterface. Liquid Instruments heeft dezelfde aanpak gekozen met zijn Moku:Lab [14], een veelzijdig apparaat met vijf tot twaalf instrumenten in één schijfvormige behuizing. Het biedt een reeks analoge in- en uitgangen en is ontworpen als een klein laboratorium. Aan de voorzijde zijn vier BNC-connectoren aangebracht (figuur 5). Het rechter tweetal levert de analoge uitgangen, met een samplingsnelheid van 1 GSa/s per kanaal bij 16bit-resolutie en een bandbreedte (–3 dB) van >300 MHz. Het linker tweetal vormt de analoge ingangen, met een bandbreedte (–3 dB) van 200 MHz in 50 Ω en een samplingsnelheid van 500 MSa/s per kanaal bij 12bit-resolutie. De interne tijdbasis biedt een nauwkeurigheid beter dan 500 ppb. Een triggeringang is ook aanwezig, samen met aansluitingen die synchronisatie van meerdere units mogelijk maken. Bedrade connectiviteit wordt geboden via een Ethernetpoort en een USB-interface, en een tweede USB-voedingspoort is beschikbaar om een tablet op te laden. Tenslotte zijn er een SD-kaartslot en een gelijkstroomingang.

Hoewel het apparaat bedrade interfaces heeft, is het eigenlijk ontworpen om te worden gebruikt in combinatie met een iPad via WiFi (802.11 b/g/n) en de bijbehorende app. Via de gebruikersinterface kunnen twaalf instrumenten worden geselecteerd, variërend van het bekende, zoals oscilloscoop en spectrumanalyzer, tot het exotische, zoals PID-regelaar en laser lock box. Het instrument functioneert ook als datalogger. De grootte van je SD-kaart is de grens voor de gegevensopslag bij maximaal 100 kSa/s bemonsteringssnelheid, terwijl de interne geheugencapaciteit het instrument geschikt maakt voor snelheden tot 1 MSa/s. Natuurlijk biedt de Moku:Lab, zoals alle moderne instrumenten, een Python API en ondersteuning voor MATLAB en LabVIEW. Ondersteund door talrijke voorbeelden



Listing 2. USBTMC in Python gebruiken om een B&K Precision-oscilloscoop aan te sturen.

```
import usbtmc
instr = usbtmc.Instrument(<VID>, <PID>, 'ABC123456')
# or, without serial number
instr = usbtmc.Instrument(<VID>, <PID>)
print(instr.ask("*IDN?"))
# returns 'BK Precision,2567B-MSO,ABC123456,5.0.1.3.9R3'
```

(**listing 3**) kan het apparaat ook snel geïntegreerd worden in een geautomatiseerde testopstelling. Het is waarschijnlijk bijzonder geschikt voor het op grote schaal testen van RF-apparaten en het sorteren van componenten.

Tijd besparen, nauwkeurigheid verbeteren

Testapparatuur vormt de ogen van de technicus, die laten zien wat er in complexe systemen gebeurt. Maar ze hebben hun beperkingen. Sporadische gebeurtenissen zijn per definitie moeilijk op te sporen, en een oplossing kan alleen worden verzonden als een mogelijke oorzaak bekend is. API-ondersteunde programmeerinterfaces op test- en meetapparatuur worden tegenwoordig steeds gebruikelijker, variërend van eenvoudig tot verfijnd. En steeds vaker is het eenvoudig te leren Python de programmeertaal bij uitstek. Als je meetprobleem moeilijk te triggeren is, steeds weer wijzigingen nodig heeft om een enkele testronde te voltooien, of gewoon saai en repetitief is (wat leidt tot bedieningsfouten), dan is automatisering niet zo ingewikkeld als je misschien dacht. ◀

vertaling: Hans Adams — 230046-03

Vragen of opmerkingen?

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



Listing 3. Een willekeurige golfvorm genereren met de Moku:Lab in Python.

```
"""pymoku example: Arbitrary waveform generator
(c) 2019 Liquid Instruments Pty. Ltd.
(shortened version for Elektor)
"""

from pymoku import Moku
from pymoku.instruments import ArbitraryWaveGen
import numpy as np

# generate a signal the the Arb Waveform Gen should generate on the output
t = np.linspace(0, 1, 100) # Evaluate our waveform at 100 points
# Simple square wave (can also use scipy.signal)
sq_wave = np.array([-1.0 if x < 0.5 else 1.0 for x in t])
# More interesting waveform. Note that we have to normalize this waveform
# to the range [-1, 1]
not_sq = np.zeros(len(t))
for h in np.arange(1, 15, 2):
    not_sq += (4 / (np.pi * h)) * np.cos(2 * np.pi * h * t)
not_sq = not_sq / max(not_sq)
# Connect to your Moku by its device name
m = Moku.get_by_name('Moku')
# Prepare the ArbitraryWaveGen instrument
i = m.deploy_or_connect(ArbitraryWaveGen)
try:
    # Load the waveforms to the device.
    i.write_lut(1, not_sq)
    i.write_lut(2, sq_wave)
    # Configure on-device linear interpolation
    i.gen_waveform(1, period=1e-6, amplitude=1, interpolation=True)
    i.gen_waveform(2, period=1e-6, amplitude=2, interpolation=False)
finally:
    m.close()
```

WEBLINKS

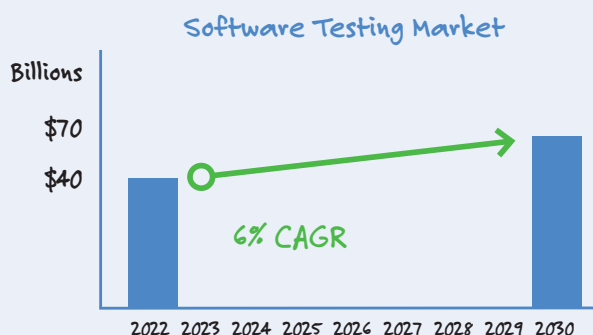
- [1] PSU Sequencer, Aim-TTi: <http://bit.ly/40jGiCQ>
- [2] EL-R serie gelijkspanningsvoedingen, Aim-TTi: <http://bit.ly/3kWZmXk>
- [3] pySerial-module: <http://bit.ly/3YgSJgZ>
- [4] Voorbeeld pySerial TCP/IP Bridge: <http://bit.ly/3kTDKem>
- [5] G. Clark, "RFC 2217 Telnet Com Port Control Option," Cisco Systems, Inc., oktober 1997: <http://bit.ly/3jcQGM4>
- [6] "Universal Serial Bus Test and Measurement Class Specification (USBTMC)," USB Implementers Forum, Inc., april 2003: <http://bit.ly/3YbJY7L>
- [7] Model 2567B-MSO, B&K Precision: <http://bit.ly/3kN3rNy>
- [8] A. Forencich, "Python USBTMC," juli 2014: <http://bit.ly/3HMLzZN>
- [9] A. Forencich, python-usbtmc, GitHub Project: <http://bit.ly/3wK4i4r>
- [10] PyUSB-software: <http://bit.ly/3WRFW3l>
- [11] libusb-software: <http://bit.ly/3wLEvIY>
- [12] "Programming Manual 2560B Series," B&K Precision, september 2022: <http://bit.ly/3WRDYjs>
- [13] "VISA Resource Syntax and Examples," National Instruments Corp., mei 2022: <http://bit.ly/3XK3okh>
- [14] Moku:Lab, Liquid Instruments: <http://bit.ly/3WT6bXb>

Test- en meetoplossingen zijn belangrijk voor en worden veel gebruikt door professionele ingenieurs, studenten en makers. Hier presenteren we enkele informatieve gegevens over test- en meetgerelateerde onderwerpen, waaronder het testen van software, systeembeveiliging, de MEMS-markt en de globale markt voor industriële sensoren.

Blijf die software testen

Werk je in de softwaretest-sector?

Zo ja, gefeliciteerd: je werkt in een gezonde sector ter waarde van meer dan \$ 40 miljard. Dat is helemaal niet verkeerd! Volgens een recent rapport van Global Market Insights zal de markt voor softwaretesten tegen 2030 naar verwachting met 6% groeien, wat betekent dat je de komende jaren waarschijnlijk behoorlijke kansen op een baan hebt (en wellicht ook zekerheid op langere termijn) [1].



Bedrijven om in de gaten te houden

- > Accenture
- > Atos SE
- > Amdocs
- > Cognizant
- > IBM
- > Infosys
- > Keysight (Eggplant)

(Bron: Global Market Insights)

Wat houdt een test in?

Black Box



- > Functionele test: test functionaliteit en gedrag van de app
- > Professionele programmeerkennis niet vereist
- > Uitgevoerd door testers
- > Toegang tot code niet vereist

White Box



- > Structurele test: test van de infrastructuur van de app
- > Professionele programmeerkennis vereist
- > Uitgevoerd door ontwikkelaars
- > Toegang tot code vereist

Wil je een portie software testen? Wil je een applicatie debuggen? Vind je veiligheid belangrijk? Kies 'black box' of 'white box'! Maar voordat je voor wit of zwart kiest [2] en gaat testen, is het geen gek idee om de volgende nuttige bronnen te raadplegen!

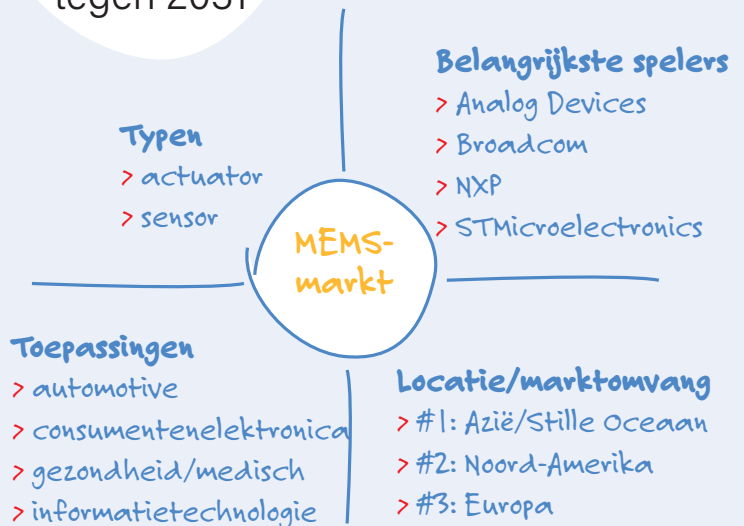
HANDIGE BRONNEN:

- > S. Cording, "Microcontrollers debuggen zonder debugger", 1 februari 2023: www.elektormagazine.nl/articles/microcontrollers-debuggen-zonder-microcontroller
- > S. Cording, "The Full Gamut of Microcontroller Debugging Techniques", 30 juni 2020: www.elektormagazine.com/mcu-debug-20
- > M. Horkan, "A Black-Box Approach to Embedded Systems Vulnerability Assessment" (Elektor Business 4/2018): www.elektormagazine.com/blackbox2018
- > L. Labbe, "De oorzaak van software bugs draadloos opsporen: circulaire buffer en webserver op de ESP32", 29 september 2022: www.elektormagazine.nl/magazine/elektor-282/61143
- > C. Valens, "Secure Communications – An Interview with Luka Matic", 6 november 2019: www.elektormagazine.com/secure-comm-19

De MEMS-markt

De markt voor micro-elektromechanische systemen (MEMS) groeit en verschillende onderzoeksbureaus melden dat de wereldmarkt tegen 2031 waarschijnlijk zal toenemen met een samengesteld jaarlijks groeipercentage (CAGR) van 8% tot 11% (volgens Verified Market Research) [3][4]. Als apparaten met zowel elektrische als mechanische functies vinden MEMS-apparaten talloze toepassingen in een breed scala van sectoren, waaronder industriële meetapparatuur, automotive, consumentenelektronica en ruimtevaart. De superkleine componenten worden gebruikt voor drukmeting, signaalverwerking, gegevensoverdracht en nog veel meer. "De sector consumentenelektronica had het grootste aandeel in 2021 en zal naar verwachting van 2022 tot 2031 groeien met een aanzienlijke CAGR voor wat betreft de marktomvang van micro-elektromechanische systemen (MEMS)", meldde Allied Market Research in september 2022 [5]. Als je geïnteresseerd bent in de MEMS-technologie, houd dan in de gaten hoe MEMS-chips op medisch gebied worden gebruikt in microchirurgische instrumenten, implantaten en micronaalden voor medicijndosering.

MEMS-markt =
+11%
tegen 2031

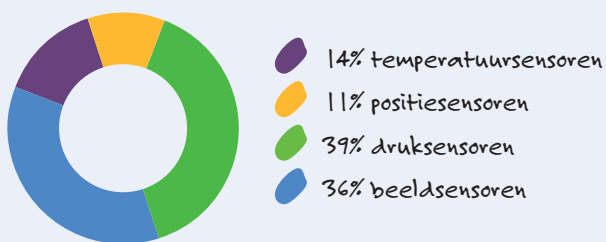


Industriële sensoren nader bekeken

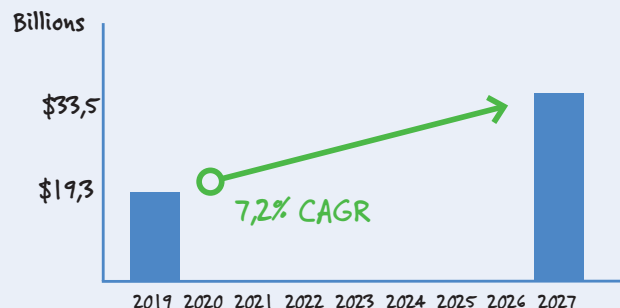
Volgens Fortune Business Insights zal de globale markt voor industriële sensoren tegen het einde van 2027 meer dan 33 miljard dollar waard zijn [6]. Deze sensoren worden gebruikt in een grote verscheidenheid aan Industry 4.0-toepassingen en zijn

onmisbaar voor meet- en testdoeleinden. Industriële sensortypen omvatten temperatuursensoren, flowsensoren, gassensoren, bewegingssensoren en positiesensoren.

Marktaandeel
sensoren 2021



Globale markt voor industriële sensoren

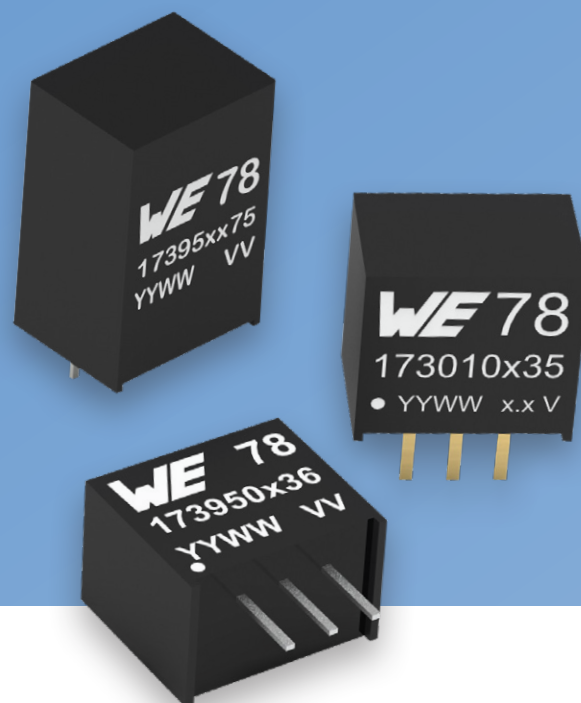


WEBLINKS

- [1] Global Market Insights, "Software Testing Market Report" 2022: <https://www.elektormagazine.com/GMI-testing-software>
- [2] L. Nguyen, "Key Differences and Similarities Between Black Box and White Box Software Testing," Orient Software, 2021: <https://www.orientsoftware.com/blog/black-box-and-white-box-software-testing/>
- [3] Verified Market Research, "Microelectromechanical System (MEMS) Market Size And Forecast," 2023: <https://www.elektormagazine.com/vmr-mems-market>
- [4] Markets and Markets, "Micro-Electro-Mechanical System (MEMS) Market," 2023: <https://www.elektormagazine.com/mandm-mems-market>
- [5] Allied Market Research, "Microelectromechanical System (MEMS) Market," 2022: <https://www.elektormagazine.com/amr-mems-2022>
- [6] Fortune Business Insights, "Industrial Sensors Market Size, Share & COVID-19 Impact Analysis," 2019: <https://www.elektormagazine.com/fortune-sensors>
- [7] Markets and Markets, "Top 10 Sensors Market by Type (2021-2026)," 2021: <https://www.elektormagazine.com/markets-top-10-sensors>

Overspanningsbeveiliging voor veilige werking

transiëntbescherming voor niet-geïsoleerde DC/DC-voedingsmodules



Timur Uludag (Würth Elektronik)

In industriële omgevingen komen veel overspanningspieken voor vanwege de uitgebreide elektrische infrastructuur. Bij het ontwikkelen van een efficiënt filter om die piekspanningen te begrenzen, moet met veel parameters rekening gehouden worden.

Veel industriële toepassingen werken tegenwoordig met logisch niveau-ingangsspanningen zoals 5 V_{DC} of lager (zie **figuur 1**). Zulke toepassingen worden vaak gevoed uit een distributiesysteem met een DC-busspanning van 24 V_{DC}. Vaak wordt gebruik gemaakt van schakelende DC/DC-converters voor het omzetten van de hogere busspanning naar het lagere spanningsniveau.

In **figuur 1** ziet u de elektrische basisstructuur van een industriële installatie. De verschillende delen van de toepassingen worden gevoed via een gelijkspanningsbus. Ter plaatse wordt elke aparte elektrische belasting via een subnet verbonden met 24 V. Niet-geïsoleerde voedingsmodules leveren de voedingsspanning voor alle subsystemen.

De bron van transiënten

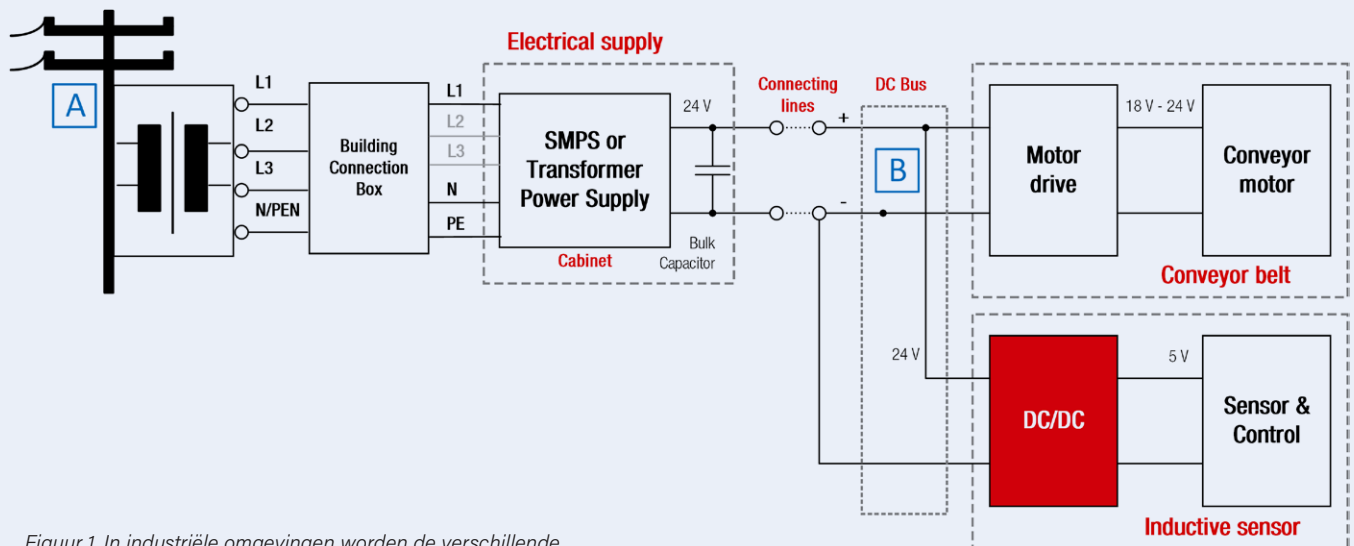
Spanningspieken (transiënten) zijn kortstondige afwijkingen van een nominale spanningswaarde, die het

toelaatbare tolerantiebereik van de nominale spanning in een elektrisch systeem overschrijden. Deze kunnen schade aan het systeem veroorzaken.

Voor een transiënt op een DC-bus kan niet één enkele oorzaak worden aangewezen. Er kan sprake zijn van een indirecte blikseminslag (**figuur 1, bij A**), of het systeem zelf kan de piek hebben veroorzaakt (**figuur 1, bij B**). Normaal gesproken loopt het toegestane spanningsbereik van de 24 V-bus in een industriële omgeving van 19,2 V tot 30 V. Bij transiënte overspanningen moeten we rekening houden met andere effecten. Als de 24V-voedingsslijn bijvoorbeeld parallel aan de besturingslijn van een frequentie-omvormer verloopt, worden de pulsen capacitief gekoppeld en oscilleert de 24 V in het pulspatroon van de omvormer. Onjuiste of ontbrekende bescherming tegen spanningspieken/transiënten leidt tot defecten door schade aan de DC/DC-converter, wat meer systeemstilstand en kosten met zich meebrengt. Voor de juiste berekening en aannames moeten we gebruik maken van een genormaliseerde transiënt, zoals beschreven in de standaard IEC 61000-4-5.

Filterconcept

In **figuur 2** ziet u het hele concept van het beschermingsfilter (groen), dat uit twee filtertrappen bestaat. Eén trap begrenst de hoge transiënte overspanning als die optreedt, dat kan met componenten zoals eenzijdige TVS-diodes (Transient Voltage Suppression).



Figuur 1. In industriële omgevingen worden de verschillende elementen van een installatie typisch gevoed via een DC-bus.

Als tweede trap wordt een passief LC-filter aanbevolen. Dit verzwakt de spanningen die hoger zijn dan de maximale werkspanning van de DC/DC-converter.

Ontwerpeisen

De belangrijkste parameter van een schakelende regelaar die we nodig hebben voor het ontwerp van een transiëntenfilter is deingangsspanning. Vaak worden twee verschillende waarden gegeven in de datasheet. Eén daarvan is de absolute maximale ingangsspanning. Als die wordt overschreden, leidt dat tot onherstelbare schade aan de voedingsmodule. De andere is de maximale werkspanning, dat is de gespecificeerde maximale ingangsspanning waarvoor de fabrikant de voedingsmodule heeft ontworpen. Voor een transiëntenbeveiliging wordt aanbevolen het beschermingsfilter zó te ontwerpen dat zelfs wanneer een transiënt optreedt, de ingangsspanning van de module nooit de maximale werkspanning overschrijdt. Voor de verdere berekeningen gebruiken we als voorbeeld een voedingsmodule in een SIP-3 behuizing (Würth Elektronik 173010535). Deze heeft een absolute maximale ingangsspanning van $V_{in} = 44\text{ V}$.

Ontwerp van een beschermingsfilter

In dit artikel gebruiken we een unidirectionele TVS-diode als beschermend element voor de ingang van de voedingsmodule. Het unidirectionele gedrag van deze diode betekent dat de V/I-karakteristiek bijna hetzelfde is als die van een zenerdiode. Daarom wordt de diode normaal gesproken in sperrichting gebruikt. Als de doorslagspanning van de TVS-diode wordt overschreden, gaat hij geleiden. Het spanningsniveau wordt dan bepaald door de stroom door de diode. Het volgende rekenvoorbeeld is een vereenvoudigde handreiking voor het dimensioneren van een filter. Met de verkregen schattingen kunnen we snel verbeteringen aanbrengen bij tests in de praktijk.

Voor een goed filterontwerp op basis van een TVS-diode hebben we de volgende parameters nodig:

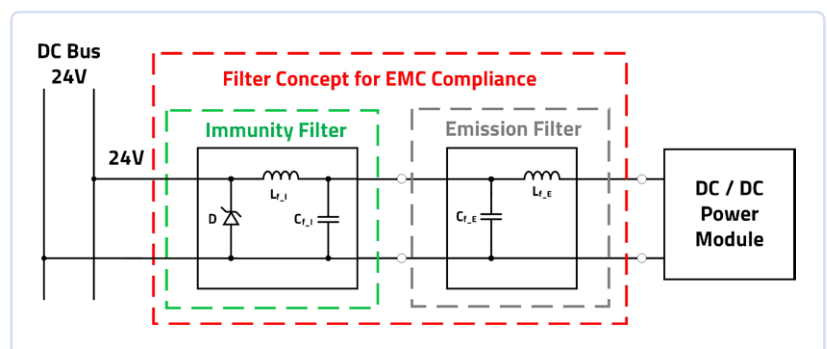
- V_{DC} : voedingsspanning van de voedingsmodule;
- V_{BR} : spanning waarbij 1 mA door de TVS-diode loopt;
- I_{Peak} : maximale piekstroom door de TVS-diode bij $V_{Clamp\ max}$;
- $P_{Diss\ max}$: maximaal toegestane dissipatie in de TVS-diode;
- $V_{Clamp\ max}$: spanning waarbij de diode de maximaal gespecificeerde stroom voert.

Eerste filtertrap

Bepaling van V_{DC}

Voor V_{DC} is de bepalende waarde de maximale DC-busspanning die aanwezig kan zijn, niet de nominale waarde van 24 V voor een 24V-bus. In een industriële omgeving is een 24VDC-bus gespecificeerd als 19,2...30 V. Het maximum is dus $V_{DC} = 30\text{ V}$. De keuze van de TVS-diode voor de volgende stappen van de berekening is gebaseerd op de verkrijgbare componenten in het productportfolio van Würth Elektronik. In de online-catalogus zien we twee mogelijke kandidaten: de 824541301 en de 824551301 [1].

Figuur 2. Filterconcept voor de EMC-compliance van Mag³C-voedingsmodules met beschermings- en emissiefilter.





Bepaling van V_{BR}

V_{BR} is gedefinieerd als de spanning waarbij een stroom van 1 mA door de TVS-diode loopt. Deze waarde, in dit geval 35,05 V, is in werkelijkheid geen vaste waarde omdat de PN-overgang een tolerantie heeft. Volgens de datasheet bedraagt de tolerantie $\pm 5\%$. We hebben dus te maken met een V_{BR} -band van 33,325 V tot 36,8025 V. In dit gebied gaat de diode geleiden met een stroom van 1 mA.

Maar we moeten ook weten tot welke waarde de transient-spanning moet worden beperkt. Dat is de parameter $V_{Clamp\ max}$.

Bepaling van $V_{Clamp\ max}$

Op deze waarde is te vinden in de datasheet. Voor de gekozen diode bedraagt deze spanning 48,4 V bij een piekstroom I_{Peak} van 31 A, dat is bij een puls van 10/1000 μs .

Tot nu toe hebben we gerekend met een ideale laboratoriumomgeving met een stabiele omgevingstemperatuur van 25 °C. Maar de werkelijkheid ziet er anders uit. Omgevingstemperaturen tot wel 55 °C zijn gebruikelijk voor elektronische componenten zoals een TVS-diode. Daarom moet in de berekening ook een temperatuurfactor worden opgenomen.

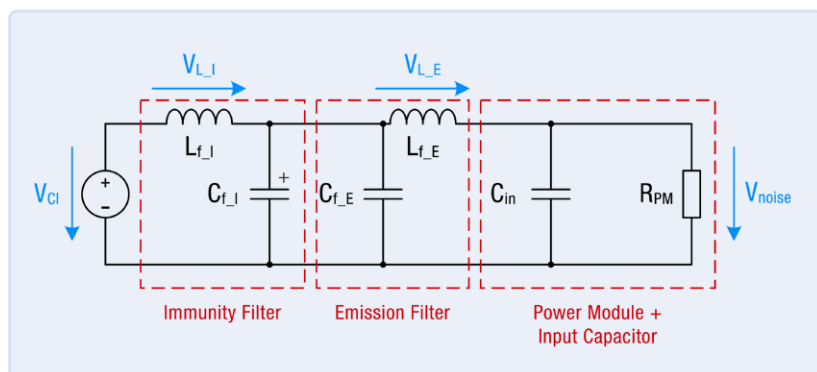
Vooral $V_{Clamp\ max}$ en het piek-pulsvermogen zijn sterk afhankelijk van de temperatuur.

In **vergelijking 1** ziet u welk effect de temperatuur op $V_{Clamp\ max}$ heeft.

$$V_{Clamp\ max}(T_j) = V_{Clamp\ max}(25^\circ C) * (1 + \alpha T * (T_j - 25^\circ C))$$

In de 'standby-situatie' loopt er vrijwel geen stroom door de TVS-diode, afgezien van de lekstroom van 1 μA als de junctietemperatuur (T_j) bijna gelijk is aan de omgevingstemperatuur. Uitgaande van een temperatuurcoëfficiënt αT voor dit type TVS van $9,9 \times 10^{-4}$ per °C, geeft dit een maximale V_{Clamp} van 49,84 V bij 55 °C. Deze waarde wordt nu het uitgangspunt voor de dimensionering van de tweede trap van het filter.

Figuur 3.
Vervangingsschema voor het berekenen van de tweede filtertrap.



Tweede filtertrap

De vraag is nu hoe we de juiste filterverzwakking moeten inschatten en hoe we de beste waarden voor de filtercomponenten kunnen bepalen. We beginnen met de verzwakking: de minimale filterverzwakking kan worden berekend met **vergelijking 2**.

$$G = 20 \cdot \log \left(\frac{V_{PM_Max}}{V_{Clamp\ max}} \right) \quad G = 20 \cdot \log \left(\frac{44V}{49,84V} \right) = -1,08dB$$

(In plaats van het symbool A (attenuation) wordt in de formule G (gain) gebruikt. Een negatieve versterking (gain) betekent een verzwakking.)

Vergelijking 2 geeft de resulterende spanning $V_{Clamp\ max}$ van de TVS-diode tijdens een spanningspiek en bij de maximale werkspanning V_{PM_Max} van de gekozen converter. We willen een filter ontwerpen zoals in **figuur 3**, waar een LC-filter moet worden toegevoegd aan de TVS-diode. De ontwerper kan een spoel kiezen en de daarbij passende condensatorwaarde uitrekenen. De filterspoel staat namelijk in serie met de toepassing, dus zijn weerstand (R_{DC}) veroorzaakt ongewenste verliezen. We moeten dus kiezen voor een spoel met een zo klein mogelijke R_{DC} -waarde, die geschikt is voor de nominale maximale uitgangsstroom van de DC/DC-converter. Voor dit voorbeeld-filterontwerp is gekozen voor een WE - PD2 (744776112) met een zelfinductie van 12 μH , een R_{DC} van 336 m Ω en een nominale stroom van 2,72 A. De DC-ingangsweerstand van de voedingsmodule kan worden bepaald met de gegeven in- en uitgangsspanning, de uitgangsstroom en de efficiëntie tijdens gebruik. Met deze gegevens kan **vergelijking 3** voor de ingangsweerstand worden uitgewerkt.

$$R_{PM} = \frac{V_{in}^2}{\frac{V_{out} \cdot I_{out}}{\eta}} = \frac{V_{in}^2}{P_{in}} \quad R_{PM} = \frac{(24V)^2}{\frac{5V \cdot 1A}{0,88}} = 101\Omega$$

In figuur 3 ziet u het vervangingsschema, waarin de TVS-diode wordt vertegenwoordigd door een vereenvoudigde spanningsbron tijdens de spanningspieken. De rest van het schema voor het EMC-model bestaat uit twee LC-filters voor transientbescherming en tegen uitstraling (EMI-onderdrukking), de ingangscondensator van de DC/DC-converter en de ingangsweerstand van de regelaar.

Omdat dit een toepassing voor binnenshuis is, is er indirecte inslagkoppeling. De volgende aannames en berekeningen zijn gebaseerd op een 8/20 μs -puls. Voor verdere vereenvoudiging kunnen we C_{f_E} en L_{f_E} weglaten, want dit filter is ontworpen voor het onderdrukken van storingen bij de schakelfrequentie

van de voedingsmodule (**figuur 4**). De voedingsmodule schakelt typisch bij 520 kHz en de piek is hier gebaseerd op 1 kHz. Wanneer we het frequentiespectrum van de puls bekijken, kunnen we zien dat de hoogste waarde van de ruisspanning optreedt bij een frequentie $f = 1$ kHz.

Om de verzwakking G van het filter te bepalen, moeten we de uitgangsspanning vergelijken met de ingangsspanning van het systeem (**vergelijking 4**).

$$\frac{V_{\text{noise}}}{V_{\text{Cl}}} = \frac{1}{Z_{\text{Lf}} + \left(\frac{1}{Y_{\text{Cf}} + Y_{\text{Cin}} + Y_{\text{RPM}}} \right)}$$

$$G = 20 \cdot \log \left(\left| \frac{V_{\text{noise}}}{V_{\text{Cl}}} \right| \right)$$

$$G_{\text{immunityfilter}} = 20 \cdot \log \left(\left| 1 - \omega^2 L_f (C_f + C_{\text{in}}) + j\omega \frac{L_f}{R_{\text{PM}}} \right| \right)$$

Met de eerder berekende DC-ingangsweerstand $R_{\text{PM in}}$ van de converter, kunnen we **vergelijking 5** gebruiken om de benodigde condensatorwaarde te berekenen:

$$C_f = \frac{1 - \left(10^{\frac{G}{20}} - \left(\omega \frac{L_f}{R_{\text{PM}}} \right)^2 \right)^{\frac{1}{2}}}{\omega^2 \cdot L_f} - (C_{\text{in}} + C_{\text{f_emission}})$$

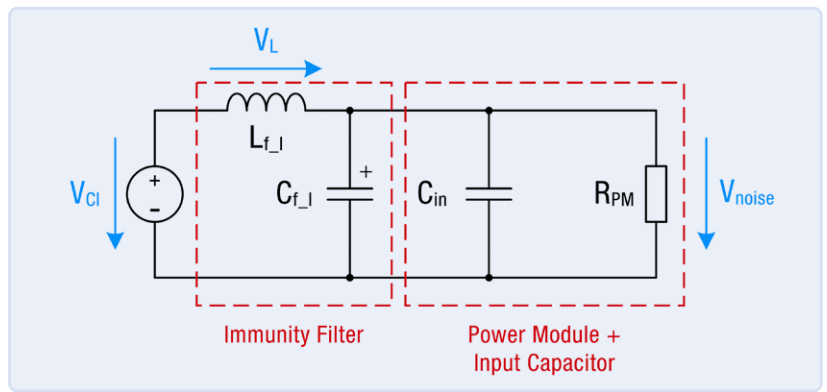
Hier is L_f de filterspoel, R_{PM} de DC-ingangsweerstand van de converter, C_{in} de ingangscondensator en $C_{\text{f_emission}}$ de EMI-filtercondensator van het PI-filter aan de ingang. De waarden zijn te vinden in de datasheet van de voedingsmodule 173010535. Aan het frequentiespectrum van de piek is te zien dat de hoogste waarde van de stoorspanning optreedt bij een frequentie van $f = 1$ kHz. Daarom wordt deze frequentie gebruikt voor de worst-case berekening. Als we uitgaan van een spoel van 12 μH , volgt uit de berekening een condensatorwaarde van $C_{\text{f_i}} = 218 \mu\text{F}$. We kijken naar de standaardwaarden voor condensatoren en komen dan uit op 220 μF (860010775018), omdat die groter is dan de berekende waarde. Een kleinere dan de berekende capaciteit zou niet voldoende verzwakking geven. Zo komen we uit op de volgende componenten (van Würth Elektronik):

TVS diode = 824541301

$L_{\text{f_i}} = 744776112$

$C_{\text{f_i}} = 860010775018$

De invloed van de temperatuur op $V_{\text{Clamp max}}$ en dus op de waarde van de filtercondensator is weergegeven in **tabel 1**.



De condensatorwaarden zijn berekend. Maar een echte condensator heeft last van tolerantie. De capaciteitswaarde kan $\pm 20\%$ variëren. Als de temperatuurafhankelijkheid van $V_{\text{Clamp max}}$ niet volledig wordt meegerekend, kan een condensator met een te lage capaciteitswaarde worden gekozen.

Transiëntbescherming en EMI

Bij het ontwikkelen van een efficiënt filter om spanningstransienten te begrenzen, moet met veel parameters rekening gehouden worden. Dat is vooral belangrijk in een industriële omgeving, omdat daar vaak transiënte overspanningen voorkomen ten gevolge van de uitgebreide elektrische infrastructuur. Het filter met transiëntbescherming maakt een efficiënte bescherming van de DC/DC-converter mogelijk en zorgt meteen voor onderdrukking van hoogfrequente stoorsignalen.

vertaling: Evelien Snel — 230180-03

Figuur 4. Vereenvoudigd vervangingsschema voor het berekenen van de tweede filtertrap.

Tabel 1.

Omgevings-temperatuur	$V_{\text{Clamp max}}$	Waarde filtercondensator
25 °C	48,4 V	178 μF
55 °C	49,84 V	218 μF



Over de auteur

Timur Uludag behaalde zijn Dipl.-Ing. (FH) graad in Mechatronica aan de University of Applied Sciences in Regensburg, Duitsland. Daarna werkte hij jarenlang als hardwareontwerper op het gebied van schakelende voedingen en analoge schakelingen. Sinds 2015 is Timur Senior Technical Marketing Manager bij de Würth Elektronik eiSos Group in de bedrijfseenheid Magl³C Power Modules. Hij is daar gespecialiseerd in de roadmap-planning en het naar de markt brengen van nieuwe voedingsmodules.

WEBLINK

[1] Würth Elektronik datasheets voor TVS-diodes:
<https://we-online.com/en/components/products/WE-TVSP>

Meetapparatuur van het merk Wiha



testers en meters voor installaties waar u op kunt bouwen

Transfer Multisort Elektronik Sp. z o.o.

Elektrische testapparatuur is essentieel voor elke technicus die niet in het donker wil werken. Het aanbod aan meetapparatuur van WIHA is nog maar het begin van het indrukwekkende aanbod van Transfer Multisort Elektronik.

Meetapparaten zijn uitrustingselementen die zowel in de onderhoudswerkplaats van een grote, moderne fabriek als in de garage van een doe-het-zelver van pas komen. Zoals de naam al doet vermoeden, worden meetapparaten gebruikt om waarden te testen die kenmerkend zijn voor elektrische apparaten en installaties, hoewel u ze ook kunt gebruiken om omgevingsomstandigheden te meten. Ze maken het veel gemakkelijker om een storing te vinden in niet-functionerende apparatuur of om de goede werking van een zojuist aangelegd circuit te controleren.

Bij het zoeken naar de juiste klasse meetapparaten [1] is het de moeite waard om het aanbod van WIHA [2] in ogenschouw te nemen. De gereedschappen van deze Duitse fabrikant staan wereldwijd bekend om hun uitstekende kwaliteit, precisie en betrouwbaarheid. Dit wordt bevestigd door de veelheid aan onderscheidingen die het bedrijf ieder jaar ontvangt, onder meer voor innovatie, design en klantvriendelijkheid. Het aanbod is nu aangevuld met meetapparatuur.

Contactloze spanningstesters en spanningsindicatoren van Wiha

Een contactloze spanningsindicator is zo ongeveer een must voor elke elektricien. Hierdoor is het gemakkelijk en eenvoudig om vast te stellen of een bepaald circuit onder

spanning staat. Metingen met dit type apparaten zijn uiterst veilig, aangezien de apparatuur de spanning op afstand kan detecteren en u het potentieel gevaarlijke gebied niet hoeft aan te raken.

Het aanbod van WIHA omvat drie spanningsindicatoren, verschillend in prijs en mogelijkheden. Het eenvoudigste apparaat heeft een enkele diode die de gebruiker informeert wanneer een spanning tussen 90 V AC en 1000 V_{AC} wordt gedetecteerd. De andere twee apparaten geven de detectie van een potentiaal aan op een LED-schaal, en de indicatie gaat altijd gepaard met een akoestisch signaal. Bovendien zijn de duurdere instrumenten uitgerust met een zaklamp. Vermeldenswaard is ook dat het gemeten spanningsbereik groter is dan bij het basismodel, namelijk van 12 V_{AC} tot 1000 V_{AC}. Het enige verschil tussen het middenklasse- en topklasse-apparaat is de ATEX-certificering. Hierdoor weten we dat het topmodel veilig kan worden ingezet op plaatsen met potentieel explosiegevaar.

Tweepolige spanningsmeters (elektrische testers)

Een ander handig apparaat in de werkplaats is de tweepuntsspanningsmeter. Deze is bijzonder handig bij het doormeten van elektrische installaties en schakelkasten.

Elk van de drie beschikbare apparaten maakt

de meting mogelijk van wisselspanning in het bereik van 12 V tot 1000 V en gelijkspanning van 12 V tot 1500 V in overeenstemming met DIN EN 612433 en VDE 0682-401, en continuïteitstests van circuits van 0 Ω tot 500 kΩ. Het grootste verschil zit tussen de middenklasse- en topklasse-toestellen. De middenklassemeter is ontworpen voor het testen van fotovoltaïsche installaties en maakt ook contactloze spanningsmeting, frequentiemeting (van 1 Hz tot 950 Hz) en weerstand (van 0 Ω tot 1999 Ω) mogelijk. Het topklasseapparaat is ontworpen voor elektromobiliteit en maakt frequentiemetingen mogelijk in het bereik van 40 Hz tot 400 Hz.

Digitale multimeter

Digitale multimeters zijn een van de meest voorkomende soorten meetinstrumenten. Zij danken hun populariteit aan de veelheid van mogelijke metingen en hun gebruiksgemak. In deze categorie toestellen heeft de Duitse fabrikant twee meters ontwikkeld die op bepaalde details verschillen. Met de eenvoudigste van de multimeters, de WIHA.45218, kunt u de continuïteit van circuits < 30 Ω, weerstanden tot 40 MΩ, spanningen in het bereik van 0 V tot 600 V (zowel met gelijkstroom als wisselstroom), stroom van 40 mA tot 10 A AC/DC en frequenties tot 5 MHz meten.



Met de tester kunt u de continuïteit controleren van circuits van 0 Ω tot 500 kΩ.



De WIHA.45218 maakt spanningsmeting in het bereik van 0 V tot 600 V mogelijk.



Met de tangmeter van Wiha kunt u stromen meten tot maximaal 400 A AC/DC.



Met de tester meet u eenvoudig en snel de status van een stopcontact.



Ideaal voor onderhouds-afdelingen.

Het apparaat heeft ook een HOLD-functie en automatische instelling van het meetbereik. De tweede multimeter, de WIHA.45215, is een iets uitgebreider apparaat waarmee u de continuïteit van circuits $< 30 \Omega$, weerstanden tot $200 M\Omega$, spanningen van 0 V tot 1000 V AC/DC, stroom van 40 mA tot 10 A AC/DC en frequenties tot 60 MHz kunt meten. Daarnaast beschikt het apparaat over de functies TrueRMS, HOLD en Min/Max. Het is ook vermeldenswaard dat contactloze spanningsmeting mogelijk is en dat u de bereiken van de multimeter handmatig of automatisch kunt instellen.

Tangmeter van Wiha

Tangmeters zijn vooral handig voor contactloze metingen van de stroomsterkte. Plaats hiervoor eenvoudigweg de te testen kabel tussen de bekken van de meter en deze geeft vervolgens de waarde van de stroom aan. Dankzij deze functie zijn apparaten van dit type behoorlijk populair geworden onder elektriciens.

Wiha heeft een professionele stroomtang in het assortiment, nl. de WIHA.45219. Met dit apparaat kunt u de continuïteit van circuits $< 30 \Omega$, spanning in het bereik van 0 V tot 1000 V bij wisselstroom en van 0 V tot 1500 V bij gelijkstroom, stroom tot maximaal 400 A AC/DC, frequentie tot 5 MHz en

weerstand tot $40 M\Omega$ meten. De meter beschikt ook over de functies TrueRMS, HOLD en Min / Max / Avg. Vermeldenswaard is dat het apparaat ook is uitgerust met een zaklamp en kan worden gebruikt voor fotovoltaïsche installaties en in de sector elektromobiliteit.

Stopcontacttester

Een interessant apparaat van het merk Wiha is de stopcontacttester 45220. Met deze meter kunt u eenvoudig en veilig verschillende soorten stopcontacten testen. Het frontpaneel heeft drie LED's die in een bepaalde configuratie oplichten, afhankelijk van de status van het stopcontact. De tester kan onder meer omgekeerde aansluiting van L/N-kabels, ontbreken van een fase of ontbreken van aarde opsporen. Dit type apparatuur kan zeer nuttig zijn bij het werk van elektriciens.

Draaiveldindicator (fasevolgordetester)

Bij driefasen-motorsystemen is de volgorde waarin de fasen zijn aangesloten uiterst belangrijk. Een fout op dit gebied kan zelfs leiden tot motorschade en ongewenste stilstand van de productielijn. Een apparaat waarmee eenvoudig de juiste aansluiting van een driefasenmotor kan worden gecontroleerd, is de fasevolgordetester. Eenmaal aangesloten zal de meter de draairichting aangeven dankzij



Het meetresultaat wordt aangegeven door een LED en een geluidssignaal.

de ingebouwde LED's en de gebruiker informeren bij een verkeerde aansluiting. Dit type draaiveldindicator dient verplichte uitrusting te zijn voor onderhoudsdiensten.

Continuïteitstester

Bij de aanleg van verschillende soorten elektrische installaties kan zelfs de beste vakman fouten maken. De continuïteitstester kan nuttig zijn bij het lokaliseren van fouten. Het apparaat is bestemd voor professionals en maakt metingen mogelijk in twee bereiken (tot 10Ω en tot 500Ω). De test is in overeenstemming met de meetnorm CAT II 400V. De continuïteit van het circuit wordt aangegeven door een LED op de meter en een luid geluidssignaal. Volgens de fabrikant is dit in een andere ruimte of zelfs op een andere verdieping te horen. ◀

230192-03

WEBLINKS

[1] Meetapparaten: https://www.tme.eu/nl/katalog/meetapparatuur_100164/p,wiha_248/

[2] WIHA-producten: https://www.tme.eu/nl/linecard/p,wiha_248/

[3] Bron van dit artikel: <https://www.tme.eu/nl/news/library-articles/page/51790/meetapparatuur-van-het-merk-wiha/>



Figuur 1. De Moku:Lab heeft twee analoge in- en uitgangen en is aan de bovenkant van de behuizing voorzien van een gleuf die dienst doet als tabletstandaard.



Automatisch testen en samenwerken aan testresultaten

Stuart Cording (voor Mouser Electronics)

Als het gaat om het testen van complexe toepassingen, zijn een paar hulpmiddelen essentieel, zoals deze samplers uit het assortiment van Mouser Electronics.

De hedendaagse toepassingen worden steeds complexer, niet alleen het ontwikkelen, maar ook het testen. Steeds vaker is testautomatisering nodig, zelfs tijdens de onderzoeks- en ontwikkelingsfase, om die hardnekkige, sporadische en ongrijpbare problemen aan te pakken. Testsessies 's nachts of in het weekend vereisen flexibele en programmeerbare test- en meetapparatuur voor het uitvoeren van een groot aantal testcases. De metingen moeten vervolgens worden opgeslagen voor evaluatie ter ondersteuning van de jacht op die vervelende bug!

Gelukkig is er sprake van een groeiend aanbod aan test- en meetapparatuur met mogelijkheden op het gebied van automatisering en het delen van resultaten. Kwalitatief hoogwaardige leveranciers blijven het aantal aangeboden functies uitbreiden en garanderen met hun certificatieschema's gekalibreerde resultaten. Dan is er een reeks nieuwe namen. Deze leveranciers richten zich vaak op het bieden van ongeëvenaarde niveaus van meetflexibiliteit in combinatie met ondersteuning voor integratie in handgebouwde HIL (hardware-in-the-loop) testoplossingen. Dit is ideaal voor onderzoeks- en ontwikkelingsteams die hun ontwerp snel moeten testen en itereren,

maar geen toegang hebben tot vijf- of zescijferige budgets voor meetinstrumenten of een professionele HIL-opstelling.

Ultieme meetflexibiliteit en headless-test

De laatste tijd zien we een aanzienlijke groei in headless-meetapparatuur, d.w.z. apparaten die voor het weergeven van meetresultaten volledig vertrouwen op laptops, computers of tablets geïnstalleerde software. Deze berusten doorgaans op FPGA (field-programmable gate array) SoC (system-on-chip) apparaten, gekoppeld aan high-speed analoge front-ends en DAC's (digital-to-analog converters). De FPGA stelt gebruikers in staat het instrument tijdens het proces opnieuw te configureren, zodat de tool bijvoorbeeld eerst als oscilloscoop, dan als signaalanalyser en vervolgens als PID-luscontroller te gebruiken is. Bovendien zijn ze dankzij de grote interne opslagcapaciteit ook ideaal voor het vastleggen van metingen, of het nu een milliseconde-burst of een registratie van enige dagen betreft.

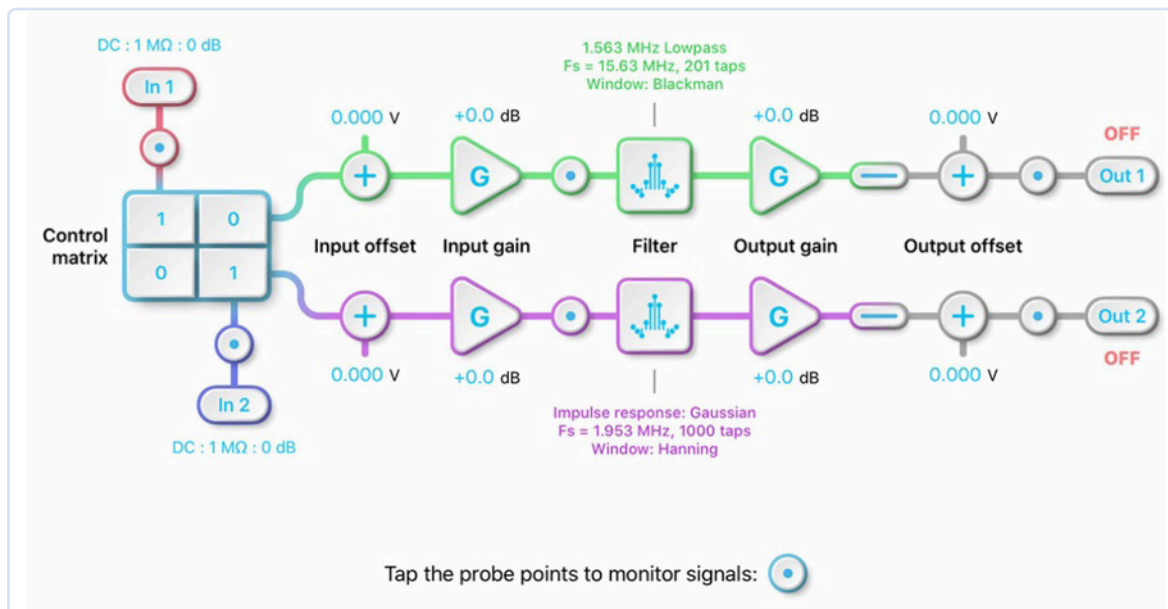
De Moku:Lab van Liquid Measurement [1] is zo'n herconfigureerbaar hardwareplatform. Dit lab biedt 12 verschillende instrumenten en wordt geleverd met software voor Windows en Mac OS desktops/laptops, en

als app voor de iPad. Dankzij zijn ronde behuizing met een diameter van slechts 20 cm (7,9") en een hoogte van 4,4 cm (1,7") past dit apparaat op iedere labtafel. De bovenkant is voorzien van een gleuf die tevens dienst doet als iPad-standaard. Gegevens, instellingen en screenshots kunnen, naast het gebruik van het interne geheugen, ook rechtstreeks worden gedeeld via Dropbox en andere clouddiensten, of worden verzonden per e-mail.

Het front is voorzien van vier verzonken BNC-connectoren. Het rechterpaar vormt de analoge uitgangen, met een bemonsteringsfrequentie van 1 GSa / s per kanaal met 16 bit resolutie en een bandbreedte (-3 dB) > 300 MHz. Het linkerpaar vormt de analoge ingangen, met een bandbreedte (-3 dB) van 200 MHz in 50 Ω en een bemonsteringsfrequentie van 500 MSa / s per kanaal met 12 bit resolutie. De interne tijdsbasis biedt een nauwkeurigheid van meer dan 500 ppb (**figuur 1**).

Aan de achterzijde zijn BNC-connectoren beschikbaar voor een externe 10 MHz-referentieklokingang en een 10 MHz-uitgang. Hiermee kunnen de signalen door meerdere Moku-apparaten worden opgevangen en met elkaar worden gesynchroniseerd. Het apparaat is tevens uitgerust met een triggeringang, evenals een ethernet-poort, een USB-interface, een USB-oplaadpoort (voor het opladen van een tablet), een SD-kaartslot en een DC-stroomingang, en heeft een connectiviteitsondersteuning via WiFi (802.11 b/g/n).

Onder de 12 ondersteunde instrumenten bevinden zich o.a. de gebruikelijke die u zou verwachten, samen met tools die tegelijkertijd gebruikmaken van de in- en uitgangen,



Figuur 2.
De Moku:Lab
FIR filterbouwer
verduidelijkt wat er
in het signaalpad
gebeurt en zorgt
voor een eenvoudige
configuratie.

waaronder een PID-controller met twee volledig configureerbare paden en een FIR filterbouwer voor eindige impulsresponsfilters met maximaal 14.819 coëfficiënten (figuur 2). Er is ook een Laser Lock Box inbegrepen. Deze maakt gebruik van de Pound-Drevel-Hall techniek [2] om de laserfrequentie te stabiliseren. Deze techniek vindt vooral haar toepassing in gravitatiegolfdetectoren, tijdmeting en atoomfysica.

Voor MATLAB, LabVIEW en Python zijn API's (Application Programming Interfaces) beschikbaar. Deze bieden talrijke

mogelijkheden voor het implementeren van geautomatiseerd testen en gegevensregistratie die allemaal worden ondersteund met voorbeelden gehost op GitHub [3]. Een robuuste en eenvoudige tweekanaals toepassing voor gegevensregistratie met Python vereist slechts enkele coderegels (figuur 3).

Lange termijn-testen bij thermische problemen

Soms wilt u de lange termijn zelfopwarming of de warmteafvoer van de behuizing van uw apparaat onderzoeken. In dergelijke

lijke gevallen kunnen infraroodcamera's een enorme hulp zijn bij het nauwkeurig lokaliseren van de warme en koude plekken. Teledyne FLIR is een begrip op het gebied van visuele temperatuurmeting. Zijn producten worden ondersteund door een grote hoeveelheid software om de gegevensanalyse, vergelijking en samenwerking te vereenvoedigen. Apparatuur zoals de MR265 [4] (figuur 4) komt met MSX (Multispectral Dynamic Imaging Technology) [5] (MSX®), die de beeldhelderheid verbetert ten opzichte van de traditionele infraroodinstrumenten voor beeldweergave. De unit bereikt dit door een warmtebeeldcamera van 160 × 120 (19.200 pixels) te combineren met een 2 MP visuele camera. Vanwege hun lagere resolutie leveren camera's met thermische beeldsensoren vaak een zeer wazig beeld.



Figuur 4: Warmtebeeldcamera's, zoals de Teledyne FLIR MR265.

```
from pymoku import Moku, StreamException
from pymoku.instruments import Datalogger
import time

m = Moku.get_by_name('Moku')

try:
    i = m.deploy_or_connect(Datalogger)

    # 100 samples per second
    i.set_samplerate(100)

    # Start data logger
    i.start_data_log(duration=10, use_sd=True, ch1=True, ch2=True,
                    filetype='bin')

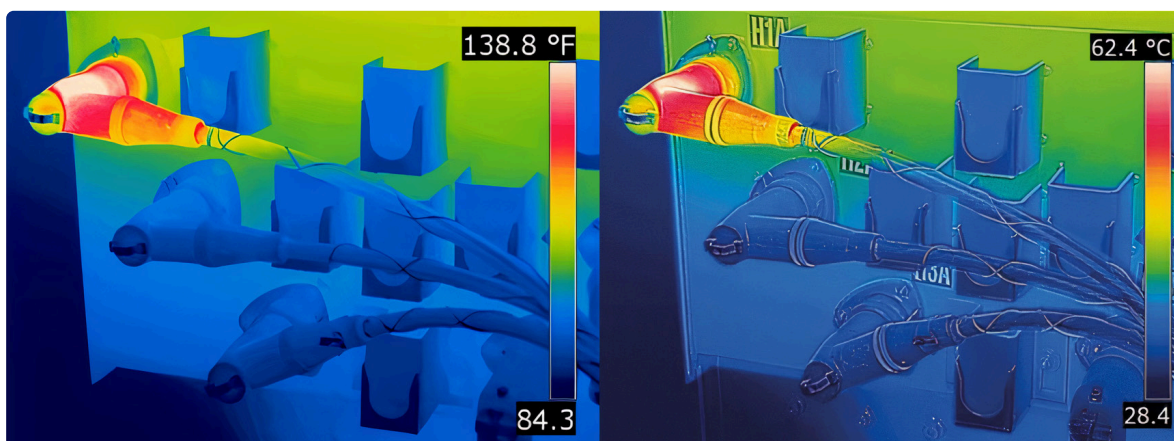
    # Upload the log file to the local directory
    i.upload_data_log()

    # Clean up
    i.stop_data_log()

except StreamException as e:
    print("Error occurred: %s" % e)
finally:
    m.close()
```

Figuur 3. Voorbeeld van een eenvoudige datalogger die de signalen van de twee analoge ingangen vastlegt, geschreven in Python voor Moku:Lab.

*Figuur 5:
Zonder MSX®
(links) ontbreken
de labels op dit
aansluitpaneel.
Met MSX® (rechts)
zijn de labels
duidelijk te zien en
wordt de algehele
beeldkwaliteit
aanzienlijk verbeterd.*



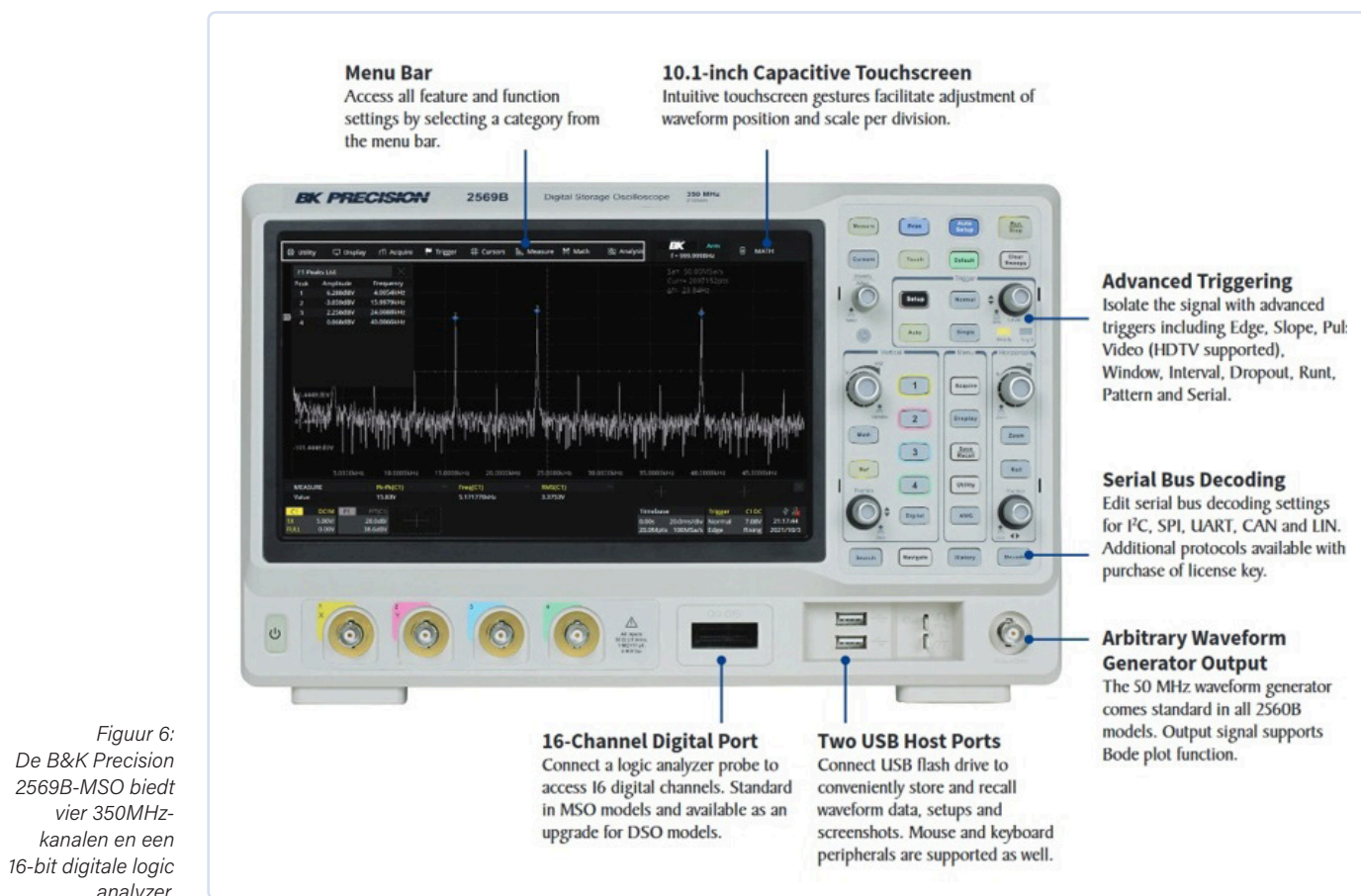
Met dergelijke hulpmiddelen gemaakte foto's kunnen zo slecht zijn, dat het haast onmogelijk is om de inhoud van de afbeelding te begrijpen zonder kennis te hebben van waar de foto is genomen. MSX® koppelt de thermische output aan een hoog-contrast beeld van de camera voor zichtbaar licht (**figuur 5**). Samen worden de zichtbare details in het beeld duidelijker en de thermische meting coherenter. Deze camera's worden ook ondersteund door de FLIR Thermal Studio Suite [6], die zowel individuele beelden als videostreams

kan verzamelen. U kunt delta's en formules toepassen voor beeldanalyse, terwijl de meetarmen worden gedefinieerd wanneer nodig. De resultaten kunnen eenvoudig worden gedeeld met collega's. Indien ondersteund, kan ook dubbele streaming worden ingeschakeld, zodat zowel zichtbaar licht als radiometrische video-opnames van het te meten doel kunnen worden opgeslagen. De MR265 biedt een 9 Hz updatesnelheid en meet temperaturen van 0 °C tot 100 °C (32 °F tot 212 °F) voor objecten op een afstand > 10 cm. Verder ondersteunt deze

unit ook vochtmetingen van 7 % tot 100 % via een ingebouwde pinloze meetsensor of afzonderlijk verkrijgbare kogel- en pinsondes. De resultaten worden weergegeven op de 2,8", 320 × 240 pixel TFT-kleurenscherm.

Geautomatiseerde metingen met traditionele labapparatuur

Hoewel zeer configureerbare, headless meetapparatuur in sommige gevallen aantrekkelijk is, kan een extra tablet of laptop voor de weergave van de resultaten een belemmering vormen. Soms is



*Figuur 6:
De B&K Precision
2569B-MSO biedt
vier 350MHz-
kanalen en een
16-bit digitale logic
analyzer.*

wat u weet gewoon beter. De B&K Precision 2560B Mixed Signal Oscilloscoop-serie [7] met een groot 10,1" capacitief touchscreen in combinatie met soft-touch- en draaiknoppen stelt in dit opzicht niet teleur. En met een diepte van slechts 111 mm (4,4") neemt hij niet veel ruimte in op de werktafel. Het topmodel 2569B-MSO biedt vier kanalen met een bandbreedte van 350 MHz en een 16-kanaals digitale poort (figuur 6). Het apparaat heeft tevens een geïntegreerde 50MHz-golfvormgenerator die een Bodediagram ondersteunt. Hierbij komen ook nog de geavanceerde activeringsopties, zoals edge, slope, pulse en video (inclusief HDTV), samen met seriële busdecodering voor de gangbare interfaces in ingebouwde systemen. Andere seriële interfaces, zoals CAN FD, FlexRay, I2S, MIL-STD-1553B en SENT, zijn verkrijgbaar als upgrade. Zowel de afstandsbediening als de gegevensvastlegging worden ondersteund via de USB- en ethernet-interfaces. Voor USB is een USBTMC (USB Test & Measurement Class) driver nodig. Dit NI-VISA product van National Instruments wordt aanbevolen [8], maar er zijn alternatieven als u bereid bent rond te kijken. GitHub biedt een selectie van projecten die USBTMC-apparaten in Python ondersteunen [9]. In combinatie

met de programmeerhandleiding van de 2560B-serie [10] is het mogelijk om scripts te ontwikkelen (figuur 7) die het vastleggen en registreren van gegevens automatiseren, evenals seriële gegevens decoderen, metingen uitvoeren of zelfs maskers definiëren voor het al dan niet aangeven of het testen van signalen aan de eisen voldoen.

Testen, testen en nog eens testen

Hoewel de meeste ontwikkelingsingenieurs hun dag het liefste aan de werktafel doorbrengen, is het opsporen van intermitterende storingen met een onduidelijke oorzaak een tijdrovende bezigheid. Gelukkig kan veel van de hedendaagse test- en meetapparatuur, zoals de hier besproken apparaten, worden gekoppeld aan software voor geautomatiseerd testen of gegevensregistratie. Dankzij de eenvoud en flexibiliteit van deze tools kunnen in enkele ogenblikken eenvoudige scripts of testprocedures worden opgesteld, vaak met open-source software, om een aannname te testen. Zodra de oorzaak van de storing is gevonden, kunnen dergelijke tests worden opgeslagen en worden gebruikt als onderdeel van CI (continuous integration) tests om ervoor te zorgen dat de storing zich niet meer

voordoeft. De resultaten zijn ook eenvoudig te delen als logbestanden via diverse clouddiensten, e-mail of cloudplatforms van leveranciers. Zo kunnen de verschillende teams deskundige collega's van over de hele wereld inschakelen om de resultaten te bekijken en te evalueren. 

230189-03



Over de auteur

Stuart Cording is een freelance journalist die o.a. voor Mouser Electronics schrijft.

Hij is gespecialiseerd in video-content en straft naar technische diepgang en inzicht. Daardoor is hij vooral geïnteresseerd in de technologie zelf en hoe deze past in eindtoepassingen, en in voorspellingen over toekomstige ontwikkelingen. Mouser Electronics is geautoriseerd distributeur van halfgeleiders en elektronische componenten en richt zich op de introductie van nieuwe producten van zijn toonaangevende partners/fabrikanten.

```
import usbtmc
instr = usbtmc.Instrument(<VID>, <PID>, <SERIAL NUMBER>)
print(instr.ask("*IDN?"))
# returns 'BK Precision,2569B-MSO,XXXXXXXXXXXXXX,5.0.1.3.9R3'
```

Figuur 7: USBTMC USB-ondersteuning is te vinden in open-source projecten. Deze code, geschreven in Python, biedt toegang tot de oscilloscoop op basis van zijn VID en PID om zijn naam en serienummer te verkrijgen.

WEBLINKS

- [1] De Moku:Lab van Liquid Measurement: <https://bit.ly/3FI0BAW>
- [2] Pound-Drevel-Hall techniek: https://en.wikipedia.org/wiki/Pound%E2%80%93Drever%E2%80%93Hall_technique
- [3] Liquid Instruments op GitHub: <https://github.com/liquidinstruments>
- [4] MR265 : <http://bit.ly/3Zb4OUK>
- [5] Verbetering van multispectrale dynamische beeldvorming: <https://flir.com/globalassets/industrial/instruments/flir-msx-tech-note.pdf>
- [6] FLIR Thermal Studio Suite: <https://flir.eu/products/flir-thermal-studio-suite/>
- [7] B&K Precision 2560 B Mixed Signal oscilloscoopserie: <http://bit.ly/40tFNW7>
- [8] National Instruments NI-VISA product: <https://ni.com/en-gb/shop/software/products/ni-visa.html>
- [9] USBTMC-apparaten in Python: <https://github.com/python-ivi/python-usbtmc>
- [10] 2560B Series programmeerhandleiding: <https://bit.ly/40lx6gi>



Bron: Shutterstock / Olha Brieva

Uit het leven gegrepen

elektronica op hoog niveau

Ilse Joostens (België)

Zo'n vijftien jaar geleden vroeg mijn toenmalige werkgever me om een agressiealarm te bouwen, omdat enkele medewerkers zich niet meer veilig voelden. Na enig, eerder teleurstellend, marktonderzoek had ik besloten om zelf maar iets draadloos in elkaar te boksen. "Gaaf je nooit lukken, koop toch gewoon iets kant en klaar in", zei een van mijn technisch georiënteerde kennissen smalend toen het onderwerp toevallig ter sprake kwam.

Het heeft me bloed, zweet en tranen gekost maar helaas voor voornoemd persoon is het me wél gelukt. Met behulp van assembler en Manchestercode nog wel. Toen kon je nu eenmaal nog geen bibliotheken met

radioprotocolen van het internet plukken en moesten de Arduino-bordjes nog de wereld veroveren. Dit illustreert perfect de huidige trend om snel resultaat te willen boeken zonder al te veel moeite te doen.

Blokkendoos

Enige jaren geleden was ik in een verhitte discussie verzeild geraakt met een pas afgestudeerde Master in de toegepaste informatica. De man had toch wel vreemde opvattingen, en het leek alsof arrogantie en neerbuigendheid deel uitmaakten van het universitair curriculum tijdens zijn opleiding. Alles wat met firmware, microcontrollers en andere toestanden op 'laag niveau' te maken had, had volgens hem geen plaats binnen de informatica maar behoorde tot de elektromechanica, een vakgebied waar hij minachtend op neerkeek. Waarschijnlijk reageerde hij zo om zijn gebrek aan kennis over het onderwerp te verdoezelen. Hij was de man van de 'OS-jes' en programmeertalen [1] op hoog niveau. Ik gun hem zijn ideeën maar het lijkt het me geen prettig vooruitzicht om bij het inschakelen van mijn koffiemachine te moeten wachten op het opstartproces van pakweg een heel Linux-systeem vooraleer het ding met het opwarmen wenst te beginnen. Het gemak dient de mens en het lijkt wel of de drang om alles op een hoger niveau te willen doen ook in de elektronica is binnengeslopen.

Ik ben nog van de 'oude stempel' en ontwerp mijn schakelingen meestal met losse onderdelen, maar ik zie op het internet en ook op Elektor Labs [2] steeds meer projecten opdruken die enkel bestaan uit een single-board computerbordje en een aantal losse kant-en-klare modules waarbij de meeste functionaliteit verzorgd wordt door software. Dikwijls bestaat die dan ook nog eens uit een bont allegaartje van stukjes open source-code en lukraak van het internet geplukte bibliotheken. Als het geheel zo ongeveer doet wat men voor ogen had, is men blijkbaar al lang tevreden – maar deze manier van werken lijkt meer op spelen met een blokkendoos dan een echt degelijk product ontwerpen. Recent heb ik nog wat zitten experimenteren met ChatGPT [3][4], een nieuwe chatbot gebaseerd op artificiële intelligentie. Blijkbaar kan dat programma op verzoek ook hele beschrijvingen van elektronische schema's en source code uitspuwen. Of het allemaal correct is, daar hebben we het raden naar maar het is best wel lekker makkelijk zo.

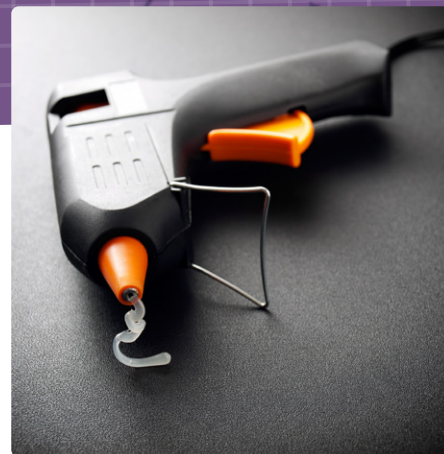
Nu ben ik helemaal niet tegen het gebruik van kant-en-klare bordjes en open source-software. Ik ben de eerste om ruiterslijk toe te geven dat ik tijdens de ontwerpfase ook wel eens terugval op bestaande software en

breakout-bordjes, maar dat is dan vooral om iets uit te zoeken, bij te leren, uit te proberen of om niet meteen een print te moeten ontwerpen wanneer het om superkleine SMD-componenten gaat. Tevens kunnen modules en breakout-bordjes educatief een grote meerwaarde betekenen zowel voor gevorderden als bij het kennismaken met de wondere wereld van de elektronica, ook al op jonge leeftijd. Getuige daarvan zijn diverse geavanceerde sets die er soms letterlijk als een blokkendoos uitzien, een hightech-versie van de traditionele elektronica-experimenteerdoois, zeg maar.

Smeltlijm

Een bevriend gitarist beweert altijd dat het leven van een muzikant bestaat uit drie procent inspiratie en zevenennegentig procent transpiratie, en volgens mij geldt dat ook voor de elektronica. U mag nog het beste idee hebben, maar als de binnenkant van uw ontwerp eruitziet als een spinnenweb van kleine printplaatjes enkel en alleen samengehouden door een berg veelkleurige draadjes, dan komt dat onprofessioneel over, laat staan dat de bij elkaar geschraapte software in alle omstandigheden betrouwbaar zal werken. Het zal niet het eerste apparaat zijn waarvan de gebruikersinterface toch niet je dat is, of waarbij de knopjes 'raar' reageren.

Op die manier doet u ook minder basiskennis op van elektronica en de kans is groot dat u ook niet (werkelijk) begrijpt wat die open source-bibliotheek doet of hoe die precies werkt. Bovendien creëert u zo afhankelijkheden want uw project staat of valt met de verkrijgbaarheid van de gebruikte modules en ook software-updates kunnen onverwacht roet in het eten gooien. De tendens om modulaire te ontwerpen wordt nog versterkt door het massale aanbod van goedkope bordjes uit het Verre Oosten. Nu ik zelf wat in de penarie zit door de componentenschaarste, heb ik ook al meerdere keren de suggestie gekregen om een bordje van Ome Ali in te zetten, maar



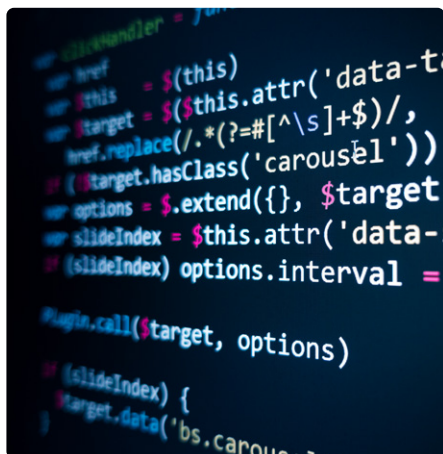
Bron: Shutterstock / Sinisa Botas

ik ben nog niet wanhopig genoeg om dat te overwegen.

Soms zijn er goede redenen om een kant-en-klare bordje te gebruiken, maar helemaal erg wordt het als commerciële bedrijven zich bezondigen aan 'spinnenwebconstructies'. Vorig jaar werd me gevraagd om even naar een medisch apparaatje te kijken met een slechte USB-connector. Toen ik de behuizing met moeite open had gekregen, bleek deze tot mijn verbazing een goedkope Arduino Nano-kloon te bevatten en een al even goedkoop bordje met een I²C-druksensor. Beiden waren met elkaar verbonden met behulp van jumperkabeltjes die waren vastgeplakt met smeltlijm of 'hot snot' [5] zoals David Jones van EEVBlog dat zo plastisch weet te verwoorden. Ook de bijhorende app was nu niet bepaald spectaculair te noemen qua complexiteit en functionaliteit. En dat moet dan een paar honderd euro kosten; puur winstbejag of een gebrek aan kennis, of beide? Ik laat het aan uw verbeelding over. ◀

230034-03

Opmerking van de redactie: Dit artikel is geschreven door een professioneel elektronicus, en we stellen het op prijs dat Ilse altijd eerlijk haar mening verkondigt. Desondanks moedigen we al onze lezers aan om hun projecten op ons Elektor Labs-platform te publiceren, ongeacht in welke staat ze verkeren. Modulaire prototypes, uit de losse pols geschetste projecten en zelfs 'kale' ideeën zijn welkom!



Bron: Shutterstock / jivacore

WEBLINKS

- [1] Wikipedia: programmeertaal: <https://nl.wikipedia.org/wiki/Programmeertaal>
- [2] Elektor Labs: <https://www.elektormagazine.nl/labs>
- [3] OpenAI: ChatGPT research release: <https://chat.openai.com>
- [4] OpenAI: Getting Started with Codex: <https://help.openai.com/en/articles/6195637-getting-started-with-codex>
- [5] EEVBlog: Sony Boombox REPAIR and Teardown: <https://youtu.be/8ToVCAhLg8A?t=487>

Energielogger

meet en registreert het stroomverbruik

Georg Luber (Duitsland)

Er zijn veel toepassingen denkbaar waarbij een continue registratie van de opwekking of het verbruik van elektrische energie zinvol is. Denk maar aan het energieverbruik van bepaalde schakelingen in huis of de energie die wordt geleverd door een balkoncentrale of een groter fotovoltaïsch systeem. Er bestaan kant-en-klare registratie-oplossingen, maar je kunt ook zelf een geschikte logger bouwen.



Figuur 1. De energielogger met meter en voedingseenheid, geïnstalleerd in een kleine plastic verdeelkast.

De hier gepresenteerde logger gebruikt een meter met een So-uitgang om het elektriciteitsverbruik te registreren. Geschikte meters zijn verkrijgbaar voor minder dan 20 euro (bijvoorbeeld bij Amazon, eBay of rechtstreeks uit het Verre Oosten). Die maken het mogelijk

het energieverbruik van aangesloten netstroomcircuits of ook de opbrengst van fotovoltaïsche systemen te meten en te 'loggen' of registreren met behulp van de hier beschreven elektronica.

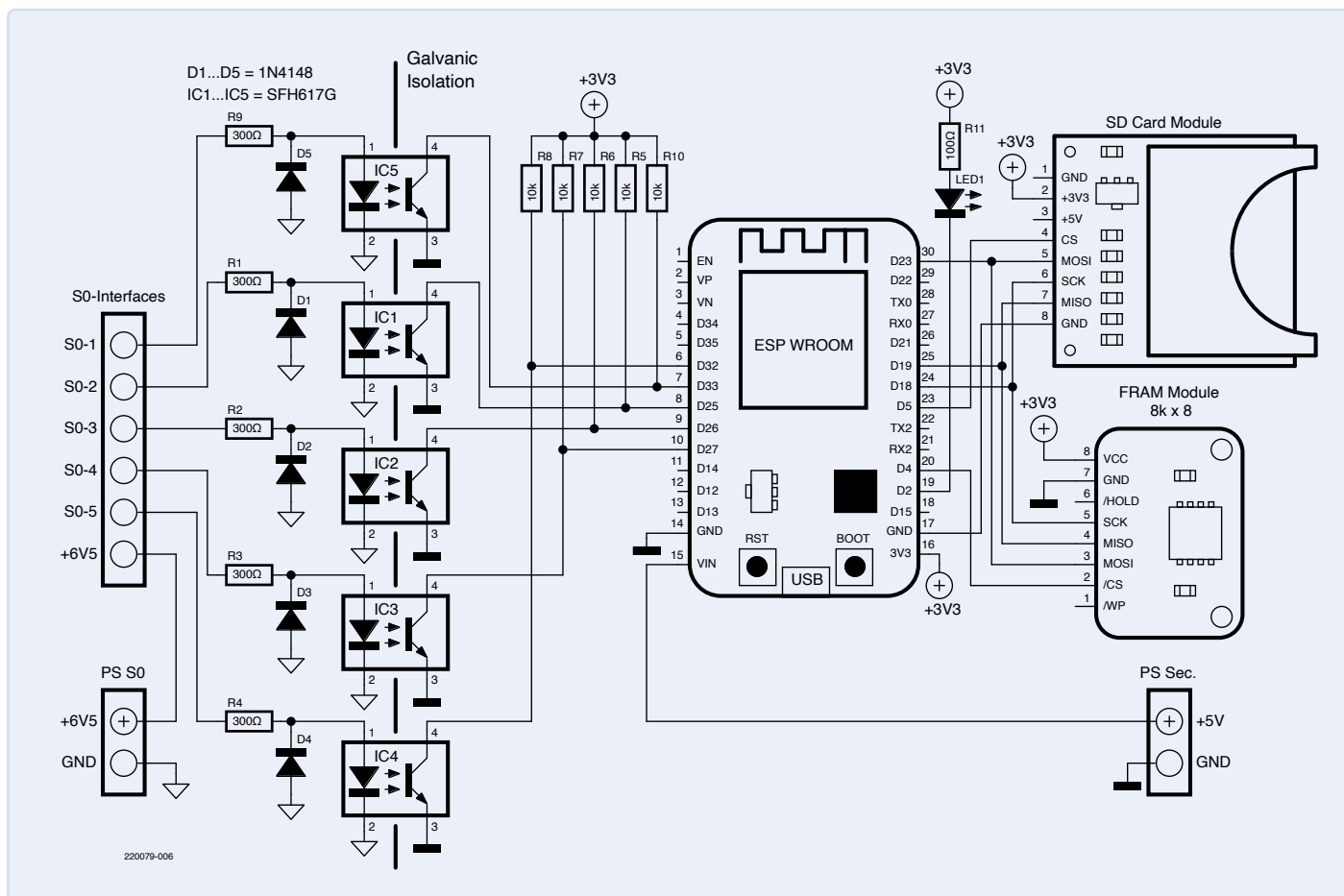
Waar zijn microcontrollers ook al weer voor uitgevonden? Voor dit soort toepassingen. Vóór het computertijdperk was het registreren van energieverbruik of -opwekking in de tijd al moeilijk, en voor particuliere toepassingen nagenoeg onmogelijk. Dankzij goedkope microcontrollers met al hun geweldige mogelijkheden is digitaal loggen tegenwoordig niet alleen geen probleem meer, maar worden zelfs fancy zaken als een WiFi-verbinding en meer mogelijk. Wat de energielogger allemaal kan, kun je nalezen in het kader **Eigenschappen**. **Figuur 1** geeft een impressie van de voltooide oplossing – een combinatie van meter en logger, inclusief voedingseenheden, geïnstalleerd in een plastic verdeelkast.

De schakeling

Dankzij de microcontroller is de schakeling van de energielogger (**figuur 2**) vrij eenvoudig. Aan de linkerkant bevinden zich een aantal optocouplers voor de galvanische scheiding, via welke de gegevens van de

Eigenschappen

- Nauwkeurig en veilig dankzij meter met So-interface en elektrische isolatie.
- Configureerbare acquisitie van maximaal vijf meters.
- Buffering van gegevens in FRAM ter bescherming van de SD-kaart.
- Registratie op SD-kaart in CSV-formaat.
- Tijdsresolutie 5 minuten.
- Per dag wordt een apart bestand opgeslagen.
- Weergave van afzonderlijke en getotaliseerde gegevens (huidige en vorige dag in kWh).
- Weergave van WiFi-veldsterkte en aantal µC-resets (startteller).
- Downloaden van gegevens via FTP.
- OTA: update-mogelijkheid via WiFi.
- Ontwikkelomgeving: VSCode met PlatformIO.



SO-interface van de meter naar de betreffende ingangen van de microcontroller worden gestuurd. Deze zijn interrupt-gestuurd om ervoor te zorgen dat er geen puls verloren gaat. Ik heb een goedkope ESP32-module gebruikt omdat die voldoende rekenkracht alsmede een WiFi-interface heeft. Aan de rechterkant zie je twee geheugenmodules – een slot voor een SD-kaart voor massaopslag en een extra FRAM-module, die de gegevens van 5 minuten tijdelijk buffert om het aantal schrijfcycli naar de SD-kaart te beperken.

Omdat de schakeling zo eenvoudig is, heb ik hem op gaatjesprint opgebouwd. Zoals je aan de twee lichtblauwe blokken kunt zien, is om veiligheidsredenen in twee aparte voedingen voorzien. Voor de SO-interfaces links is 6,5 V voldoende. De rest van de schakeling wordt gevoed door een 5V-voeding (rechts). Een belastbaarheid van 0,5 A is voor beide genoeg. Met het oog op de elektrische isolatie mogen de beide GND-lijnen van de voedingen in geen geval met elkaar worden verbonden. Bovendien moet onder de optocouplers – dus tussen hun in- en uitgangen – al het koper over een afstand van ten minste 4 mm worden verwijderd.

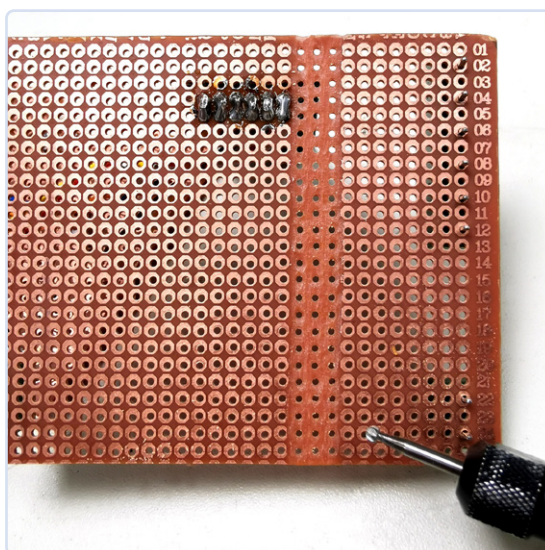
Figuur 3 toont wat de bedoeling is.

De elektrische isolatie door middel van twee afzonderlijke voedingen maakt het ook mogelijk dat de ESP32 wordt aangesloten op de USB-interface van een PC (bijvoorbeeld om toekomstige updates naar de geïnstalleerde controller te sturen). In plaats van twee voedingen zou een oplossing met slechts één voeding

met een hogere stroombelastbaarheid plus een isolerende DC/DC-converter en een spanningsregelaar ook mogelijk zijn geweest, maar dat zou de zaken er niet echt eenvoudiger of goedkoper op maken.

Gegevensopslag en -transmissie

Zoals al opgemerkt, is het gebruik van twee geheugenmodules geen overbodige luxe. De SO-interfaces kunnen soms meerdere pulsen per seconde leveren.



Figuur 2. De schakeling van de energielogger is tamelijk eenvoudig.

Figuur 3. Het koper aan de onderkant van de gaatjesprint tussen de in- en uitgangen van de optocouplers is met een klein mesje verwijderd.

Figuur 4. De gegevensstructuur wordt zichtbaar bij het importeren van een CSV-bestand in een spreadsheet.

	A	B	C	D	E
1	/2021/07/31.csv - 00:00				
2	426572	415390	132309	129324	218375
3	05:55				
4	>				
5	426572	415390	132309	129324	218375
6	426574	415391	132309	129324	218375
7	426575	415393	132309	129324	218375
8	426575	415393	132309	129324	218375
9	426576	415394	132309	129325	218375
10	426577	415396	132310	129326	218377
11	426579	415398	132310	129326	218378
12	426583	415401	132310	129326	218380
13	426587	415405	132310	129326	218382

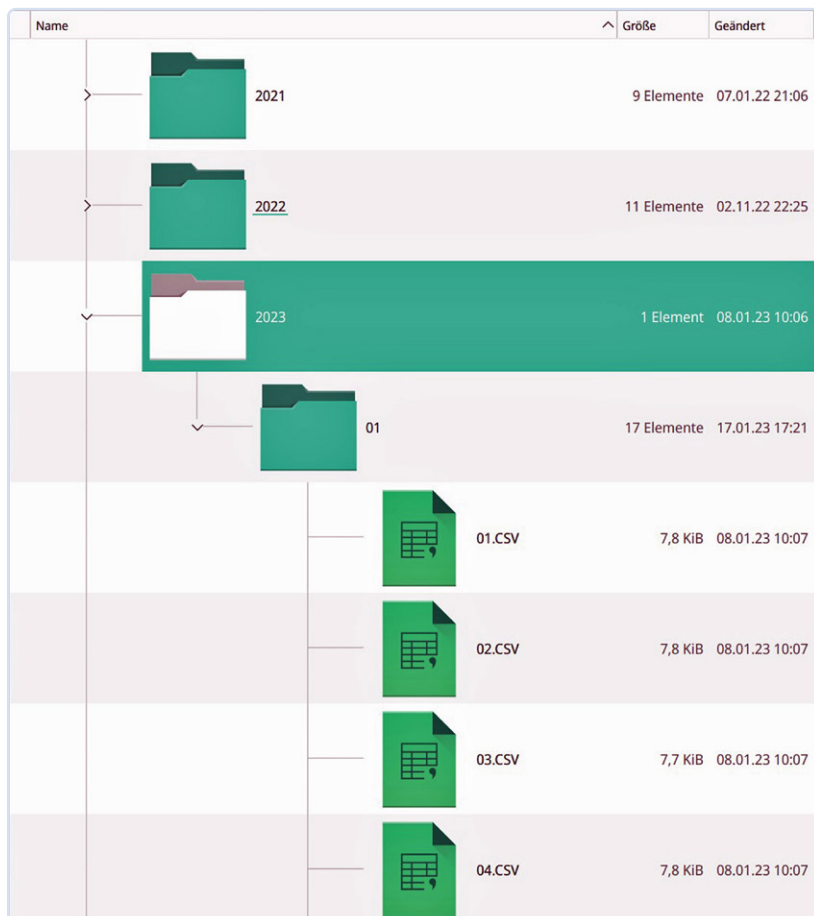
Als de meetwaarden in het interne geheugen van de microcontroller zouden worden verzameld, zouden die bij een reset verloren kunnen gaan. Aan de andere kant zou het onmiddellijk wegschrijven naar een SD-kaart de levensduur van de kaart enorm verkorten. Met slechts één waarde per seconde zouden er 31,5 miljoen schrijfcycli per jaar plaatsvinden. De geheugencellen van een SD-kaart bereiken echter het einde van hun levensduur na slechts 1.000 tot 3.000 schrijfcycli. Zelfs een kaart met een hoge capaciteit zou dus vrijwel zeker gebreken vertonen voordat een jaar is verstreken.

Om dit te voorkomen, is in een stabiele buffer voorzien. De hier gebruikte externe FRAM-module heeft een opslagcapaciteit van slechts 8 kB, maar dat is ruim voldoende voor heel wat waarden.

Het grootste voordeel van FRAM is dat het minstens 10^{10} maal beschreven kan worden, afhankelijk van de fabrikant – typisch is zo'n geheugen zelfs bestand tegen quadriljoenen schrijfcycli. Elke vijf minuten worden de in FRAM verzamelde gegevens overgebracht naar de SD-kaart. Dit betekent iets meer dan 100.000 schrijfbewerkingen per jaar, en aangezien een SD-kaart veel geheugencellen heeft, kan hij zonder problemen een flink aantal jaren worden gebruikt. Bovendien wordt het aantal herstarts van de logger geregistreerd in het FRAM, zodat je op elk moment kunt nagaan hoe vaak een reset is uitgevoerd. Elke dag wordt een nieuw bestand op de SD aangemaakt, waarin de gegevens van die dag worden opgeslagen. Om geheugenruimte te besparen wordt de opname alleen gestart als een van de meterwaarden sinds middernacht is veranderd. Als de energielogger wordt gebruikt om het energieverbruik te registreren, is deze functie eigenlijk overbodig. Als echter de energieproductie van een PV-systeem wordt bewaakt, is het nuttig dat de registratie pas begint na zonsopgang (dus zodra de eerste elektriciteit wordt opgewekt).

Gegevensstructuur en meer

Figuur 4 toont een deel van een spreadsheet waarin de gegevens in CSV-formaat zijn geïmporteerd. Cel A1 bevat de datum en het tijdstip waarop het bestand is aangemaakt. Rij 2 bevat vervolgens de laatste waarden van de vorige dag als nieuw beginpunt. De eerste verandering van een meterwaarde vond plaats om 05:55 (cel A3). Vanaf rij 5 volgen de nieuwe gegevens van deze dag. De kolommen bevatten de gegevens van de afzonderlijke meters. Elke rij heeft een tijdsverschil van vijf minuten met de volgende rij. De gegevens worden elke dag tot middernacht geregistreerd. Elke dag wordt een nieuw bestand aangemaakt. De hoeveelheid energie kan dan worden berekend uit het verschil tussen het aantal telpulsen met behulp van de puls waarde van de meters (bijvoorbeeld 0,5 Wh/puls). Het CSV-formaat is zuinig en geschikt om te importeren in een spreadsheet zoals Excel, OpenOffice/LibreOffice Calc of Numbers op de Mac. De gegevens kunnen dan naar wens verder worden verwerkt. Op de SD-kaart worden de gegevens opgeslagen in een tijdgebaseerde mappenstructuur. Op het hoogste niveau zijn er mappen voor elk jaar, die elk submappen bevatten voor de maanden, en deze bevatten dan de bestanden voor de afzonderlijke dagen. **Figuur 5** toont de structuur.



Figuur 5. De mappenstructuur: jaar, maand en dag.

In dit voorbeeld worden de waarden van vijf meters als gehele getallen geregistreerd (zie figuur 4). Dankzij het CSV-formaat is hiervoor relatief weinig geheugen nodig. Dit resulteert in enkele kilobytes per meter per dag.

De overdracht van gegevens van de logger naar een PC gebeurt via FTP. Gangbare FTP-clients zoals FileZilla zijn hiervoor geschikt. Aangezien de logger via WiFi in het thuisnetwerk is geïntegreerd, is deze methode waarschijnlijk het eenvoudigst.

Ik heb de broncode voor de ESP32 gemaakt met behulp van de PlatformIO IDE voor VSCode [1]. De code kan gratis worden gedownload van de webpagina bij dit artikel [2].

De volgende gegevens moeten worden aangepast in het bestand `main.cpp`: te beginnen bij regel 30 moeten de SSID en het wachtwoord van het WiFi-netwerk worden ingevoerd. De gebruikersnaam en het wachtwoord voor FTP worden opgegeven in regel 813. Standaard wordt "esp32" ingevoerd als gebruikersnaam en wachtwoord. Natuurlijk heeft de FTP-cliënt ook het IP-adres van de energielogger nodig. Dit adres wordt na een reset uitgevoerd via de USB-interface. Idealiter configureer je je router echter zo dat de energielogger altijd hetzelfde IP-adres krijgt toegewezen.

Ingebouwde webinterface

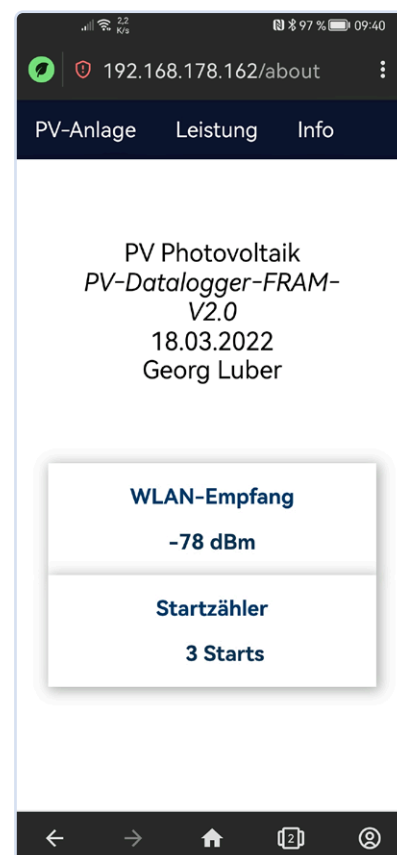
De actuele gegevens en de somwaarden van deze en de vorige dag worden weergegeven met behulp van de geïntegreerde webserver. Deze server bevat twee pagina's. Eén pagina toont het huidige vermogen per meter en het totaal van alle aangesloten meters. Daarnaast wordt de totale energie van deze en de vorige dag weergegeven in kWh. De andere pagina toont de veldsterkte van het WiFi netwerk en het aantal herstarts van de logger.

Figuur 6 en **figuur 7** zijn screenshots van een smartphone. Zoals je verderop zult merken, gebruik ik de energielogger niet om het verbruik te meten, maar om mijn PV-systeem met vijf omvormers te monitoren. Daarom heb ik de labels van de afzonderlijke waarden aangepast aan mijn specifieke situatie. Maar je kunt dit gemakkelijk in de software veranderen als dat nodig is. Wanneer je in de browser verbinding maakt met het IP-adres, krijg je het scherm van figuur 6. Door "/" about" aan het adres toe te voegen of door op de knop *Info* te klikken, verschijnt figuur 7.

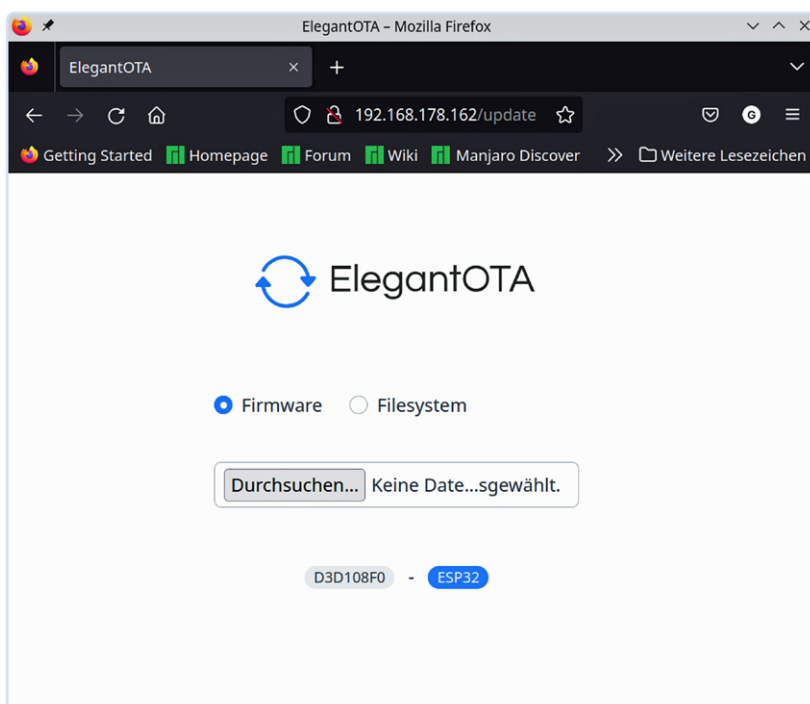
Nog een opmerking over de WiFi-verbinding: als je de



Figuur 6. De geïntegreerde webinterface geeft de meetresultaten weer.



Figuur 7. Als je "/about" achter het IP-adres plakt of op Info klikt, krijg je deze informatie.



Figuur 8. Updates van zowel de firmware als de bestandsstructuur zijn mogelijk via WiFi.

Figuur 9. Vier meters van het type DDS5188 in bedrijf.



energielogger in een metalen behuizing installeert, zul je vrijwel zeker ontvangstproblemen krijgen.

OTA

Zodra de energielogger in een behuizing is ingebouwd en geïnstalleerd, zul je de elektronica waarschijnlijk niet meer willen verwijderen bij een eventuele softwarewijziging. Daarom kan de datalogger 'Over The Air' (dat wil zeggen draadloos) worden bijgewerkt. Zowel de volledige software als het bestandssysteem – de webpagina's met JavaScript en het CSS-bestand – kunnen dus indien nodig opnieuw worden geladen. Als je dat wilt doen, moet je het IP-adres invullen met daaraan toegevoegd "/update". **Figuur 8** toont hoe dit er in mijn geval uitziet. Ik heb de OTA-mogelijkheid geïmplementeerd met behulp van de bibliotheek van Ayush Sharma [3].

Een eenvoudige schakeling

De schakeling is zo eenvoudig dat een onderdelenlijst eigenlijk overbodig is. Naast het ESP32-ontwikkelboard, dat overal verkrijgbaar is, zijn de belangrijkste onderdelen een SD-kaartslot, dat te vinden is op goedkope breakout-boards, en de FRAM-module – die ook als breakout-board kan worden verkregen. Aangezien die breakout-boards met een verschillende pinning worden vervaardigd, moet je bij het aansluiten niet letten op de pinnummers, maar op de pinbenamingen. Voor de SD-kaartmodule kan MOSI worden verbonden met DI of SI en MISO met DO of SO. De meters kunnen elk goedkoop model zijn voor DIN-rail montage met een So-interface. Het door mij gebruikte type DDS5188 (**figuur 9**) is gemakkelijk verkrijgbaar en zeer goedkoop.

Hoewel het voor zich spreekt dat je de nodige voorzichtigheid moet betrachten bij de omgang met netspanning, is een opmerking op zijn plaats: voor werkzaamheden aan de zekeringkast gelden in verschillende landen verschillende wetten. In Duitsland bijvoorbeeld moet je daarvoor bij de netbeheerder geregistreerd staan als gediplomeerd elektricien. Ook werkzaamheden aan op de zekeringkast aangesloten externe onderverdelers zijn alleen toegestaan voor een geschoolde elektricien met de juiste opleiding en kennis. ◀

220079-03

Over de auteur

Georg Luber is gediplomeerd elektricien die elektrotechniek heeft gestudeerd en vele jaren heeft gewerkt op het gebied van elektrische veiligheid en elektrische installatie. Hij werkte nationaal en internationaal mee in normalisatiecommissies (DKE/VDE, CENELEC, IEC en ISO). Een ander zwaartepunt van zijn werk was gebouwautomatisering, met name KNX. Georg Luber is betrokken bij de ontwikkeling van software- en elektronica-projecten op dit gebied.

Vragen of opmerkingen?

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



Gerelateerde producten

- **ESP32-DevKitC-32D (SKU 18701)**
www.elektor.nl/18701
- **PeakTech 4350 stroomtang (SKU 18161)**
www.elektor.nl/18161



WEBLINKS

- [1] PlatformIO voor VSCode: <https://platformio.org/platformio-ide>
- [2] Webpagina bij dit artikel: <http://www.elektormagazine.nl/220079-03>
- [3] OTA-bibliotheek van Ayush Sharma: <https://github.com/ayushsharma82/ElegantOTA>
- [4] ESP32-projecten: <https://randomnerdtutorials.com/projects-esp32>

Montage van de M.A.R.S. Rover Kit van 4tronix

Clemens Valens (Elektor)

De M.A.R.S. rover van 4tronix is geïnspireerd door de Mars-rovers Curiosity en Perseverance van NASA, en is een autonoom voertuig voor ruw terrein op onze eigen planeet in plaats van op Mars. M.A.R.S. verwijst dus niet naar de rode planeet, maar is een acroniem voor Mobile Autonomous Robotic System.

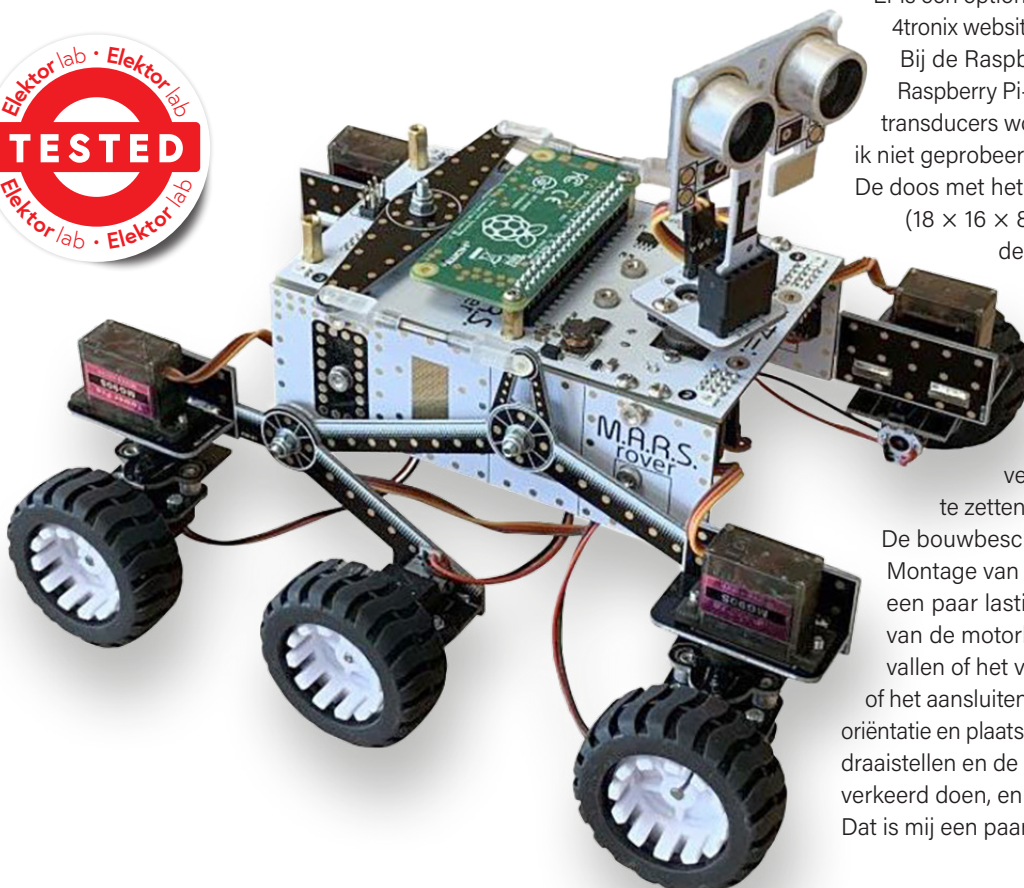
De M.A.R.S. rover wordt geleverd als bouwpakket, maar er hoeft niets gesoldeerd te worden. De kit is helemaal compleet, inclusief gereedschap, hoewel ik de voorkeur gaf aan mijn eigen schroevendraaiers, omdat die beter pasten. Er zijn twee versies van de kit: Raspberry Pi Zero en BBC micro:bit. Ik heb de Raspberry Pi-versie geprobeerd. Merk op dat de Raspberry Pi Zero of BBC micro:bit niet inbegrepen zijn, evenmin als de vier AA-batterijen. Voor de Raspberry Pi heb je ook een microSD-kaart nodig.

Er is een optioneel toetsenbordje beschikbaar op de 4tronix website [1], maar dat heb ik niet geprobeerd.

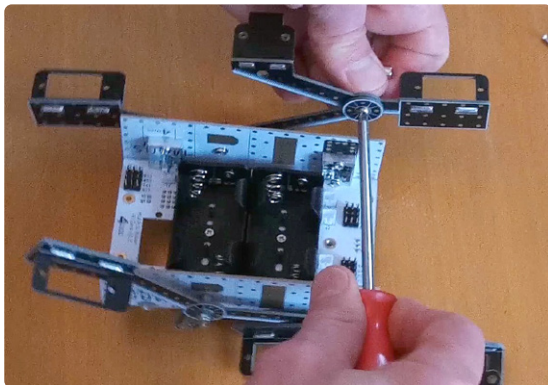
Bij de Raspberry Pi-versie van de rover kan een Raspberry Pi-camera op de mast voor de ultrasone transducers worden gemonteerd, maar ook dat heb ik niet geprobeerd.

De doos met het bouwpakket is verrassend compact (18 × 16 × 8 cm) maar vrij zwaar (600 g) en tot de rand gevuld met plastic zakjes met onderdelen. Een handleiding is niet inbegrepen. Gedetailleerde montage- en programmeer-instructies zijn beschikbaar op de website van 4tronix, dus je hebt een computer met internet-verbinding nodig om de rover in elkaar te zetten.

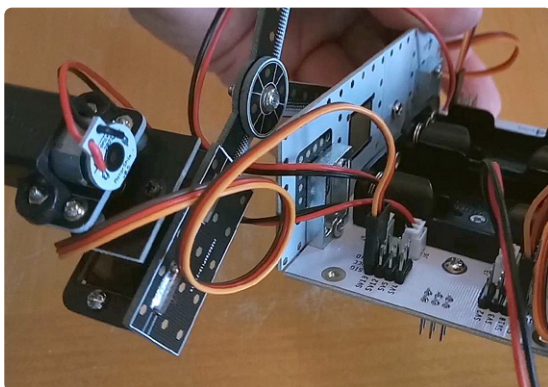
De bouwbeschrijving is duidelijk en gedetailleerd. Montage van de rover is niet moeilijk, maar er zijn een paar lastige dingen zoals het vastschroeven van de motorbeugels zonder dat de moeren eruit vallen of het vastzetten van sommige borgmoeren of het aansluiten van de motoren. Let ook goed op de oriëntatie en plaatsing van bepaalde onderdelen zoals de draaistellen en de servo's, want je kunt het gemakkelijk verkeerd doen, en dan moet je het nog een keer doen. Dat is mij een paar keer overkomen.



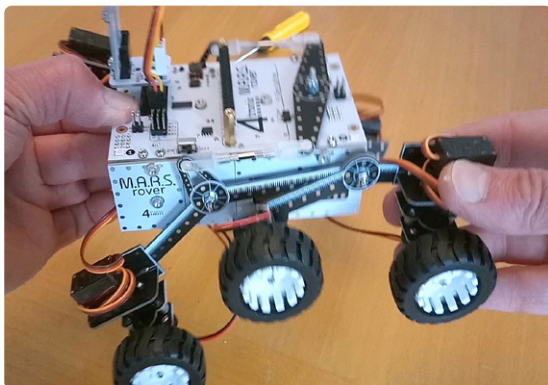
► *Figuur 1. Gebruik je eigen schroevendraaiers.*



► *Figuur 2. Alle servo-connectoren zijn op dezelfde manier georiënteerd, zijn duidelijk gelabeld en de servodraden hebben een kleurcode. Bruin is GND, geel is SIG (signaal).*



► *Figuur 3. Het draaistel op de tuimelarm moet vrijwel zonder wrijving bewegen.*



Een paar montagetips

Het in elkaar zetten van de rover kostte me ongeveer twee uur door een paar fouten die ik maakte. Hier zijn een paar tips:

- Gebruik een paar echte schroevendraaiers (**figuur 1**).
- Drie 18 mm lange schroeven gaan door printplaten. Zorg ervoor dat je ze goed vastdraait voordat je het onderdeel aan de andere kant met een borgmoer vastzet, omdat het anders los kan raken wanneer je de moer moet bijstellen om de betreffende onderdelen soepel te laten bewegen.
- Alle servo-connectoren zijn in dezelfde stand gemonteerd. Ze zijn duidelijk gelabeld, en de servodraden hebben een kleurcode (**figuur 2**). Bruin is GND en geel is SIG (signaal). Het is handig om het nummer van elke servo (SVx) te noteren voor als je het *servoTest*-programma uitvoert.
- De mast van de ultrasone transducer kan tegen de servodraden vastlopen als je die niet ver genoeg naar beneden hebt gevouwen.
- Na het aansluiten van de servo's kun je de draden ervan om de servobehuizing wikkelen zodat het er wat netter uit ziet. Laat daarentegen de draden van de wielmotoren vrij om te voorkomen dat ze de bewegingen van de tuimelarmen blokkeren (**figuur 3**).
- Het is niet nodig om de uitstekende uiteinden van de servo-armen af te snijden, dat is vooral een cosmetische ingreep.

Nu komt het brein

Na het in elkaar zetten van de rover heb ik er een Raspberry Pi Zero 2 W op gemonteerd. Ik gebruikte een microSD-kaart met het Raspberry Pi Buster-OS geïnstalleerd. Dat was al in een ander project gebruikt, dus de Raspberry Pi kon meteen verbinding maken met mijn WiFi-netwerk, en SSH-toegang was ook ingeschakeld. Om de batterijen te sparen, voedde ik de Raspberry Pi via USB bij het downloaden en installeren van de software om de rover te laten functioneren. Nogmaals, de instructies over hoe dit te doen zijn duidelijk en gedetailleerd, maar staan alleen online [2].

Ik heb ook een Raspberry Pi Zero geprobeerd in plaats van een Zero 2 W en vond die op dezelfde manier werken. Het voordeel van het gebruik van een Zero 2 is natuurlijk dat deze veel sneller opstart.

De Raspberry Pi voedt de rest van de rover niet – daar zijn de batterijen voor nodig. Als je dus een foutmelding ziet over de I²C-bus bij het uitproberen van de voorbeelden, maar je zeker bent dat je die op de Raspberry Pi hebt ingeschakeld, controleer dan de aan/uit-schakelaar en de batterijen. Een blauwe LED geeft aan of de rover is ingeschakeld of niet.



Ik heb een stroomverbruik gemeten van ongeveer 400...500 mA wanneer de rover rijdt met het *motor-Test*-script, en ongeveer 200 mA in rust.

Programmeren van de rover

Bij het ontwikkelen van eigen programma's voor de rover [3] is het waarschijnlijk het beste om dat zoveel mogelijk te doen terwijl de Raspberry Pi via USB gevoed wordt om de batterijen te sparen, en de rover alleen aan te zetten als je iets wilt uitproberen.

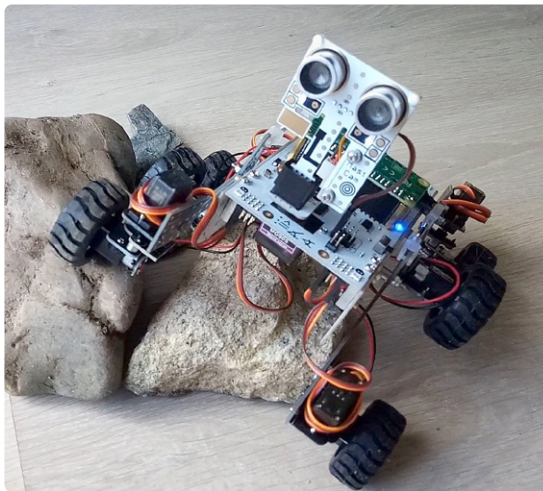
Ik heb alle voorbeelden van 4tronix [2] geprobeerd en vond alles probleemloos werken. Ik heb zelf niets geprogrammeerd. Merk op dat om onduidelijke redenen het gebruik van de RGB-LED's van de rover beheerdersrechten vereist, dus elk script dat ze gebruikt moet worden uitgevoerd met `sudo`.

De rover is bij het rijden over obstakels werkelijk indrukwekkend (**figuur 4**). Hij gaat niet erg snel, maar dankzij de tuimelarmen kan de robot obstakels overwinnen alsof die er niet zijn.

Bijna niet te stoppen

De 4tronix M.A.R.S. rover is een hoogwaardige kit die er behoorlijk *cool* uitziet als hij eenmaal in elkaar is gezet, dankzij de decoratieve opdruk en printplaten. De afmetingen bedragen 20 × 18 × 16 cm, dus een beetje groter dan de doos waarin hij wordt geleverd. Het in elkaar zetten van de robot is simpel, want alles past perfect (wat zelden voorkomt) en de instructies zijn duidelijk en gedetailleerd. Het gebruik is ook eenvoudig, en de voorbeelden en testprogramma's kunnen zonder problemen worden geïnstalleerd en uitgevoerd.

Eenmaal gemonteerd met de software geïnstalleerd, is de rover gebruiksklaar en bijna niet meer te stoppen.



Figuur 4. Na de montage is de M.A.R.S. rover bijna niet meer te stoppen.

Enige programmeerkennis is natuurlijk vereist, maar dat is toch waarom je de kit hebt gekocht, nietwaar?

Uitbreidingen

De hoofdprint heeft vier extra montagegaten die kunnen worden gebruikt om uitbreidingen te bevestigen, en er zijn nog voldoende aansluitingen voor het toevoegen van meer servo's (terzijde: het zakje met reserveonderdelen in mijn kit bevatte een extra servo).

Ik heb geen schema gevonden, dus ik weet niet of de servoconnectoren voor iets anders gebruikt kunnen worden. Er zijn ook twee I²C-uitbreidingsconnectoren beschikbaar waarop je bijvoorbeeld sensoren kunt aansluiten. ◀

220107-03



Gerelateerde producten

> **4tronix M.A.R.S. rover (SKU 19996)**
www.elektor.nl/19996

WEBLINKS

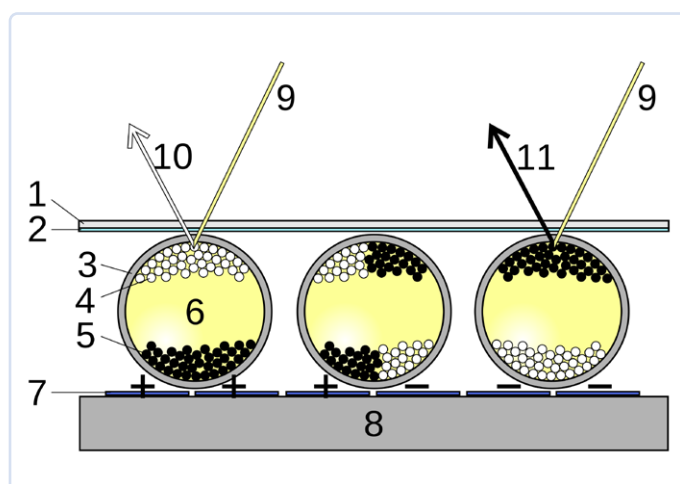
- [1] M.A.R.S. rover montage: <https://4tronix.co.uk/blog/?p=2112>
- [2] M.A.R.S. Rover programmeren (Raspberry Pi Zero (incl. v2)): <https://4tronix.co.uk/blog/?p=2409>
- [3] M.A.R.S.-Rover op GitHub: <https://github.com/4tronix/MARS-Rover>
- [4] "Assembling the 4tronix M.A.R.S. Rover Kit" – deze bespreking op video: <https://youtu.be/Np8ZQQd85oc>

Parkeerschijf met e-papier display

innovatief digitaal alternatief

Antonello Della Pia (Italië)

Een parkeerschijf is een traditioneel accessoire dat de meeste automobilisten kennen. Dit project presenteert een digitale versie op basis van een e-papier display. Het ontwerp heeft speciale functies, zoals het instellen van de aankomsttijd met één knop, een bericht in vier talen naar keuze en indien gewenst weergave van de huidige tijd en datum, de omgevingstemperatuur en het batterijniveau.

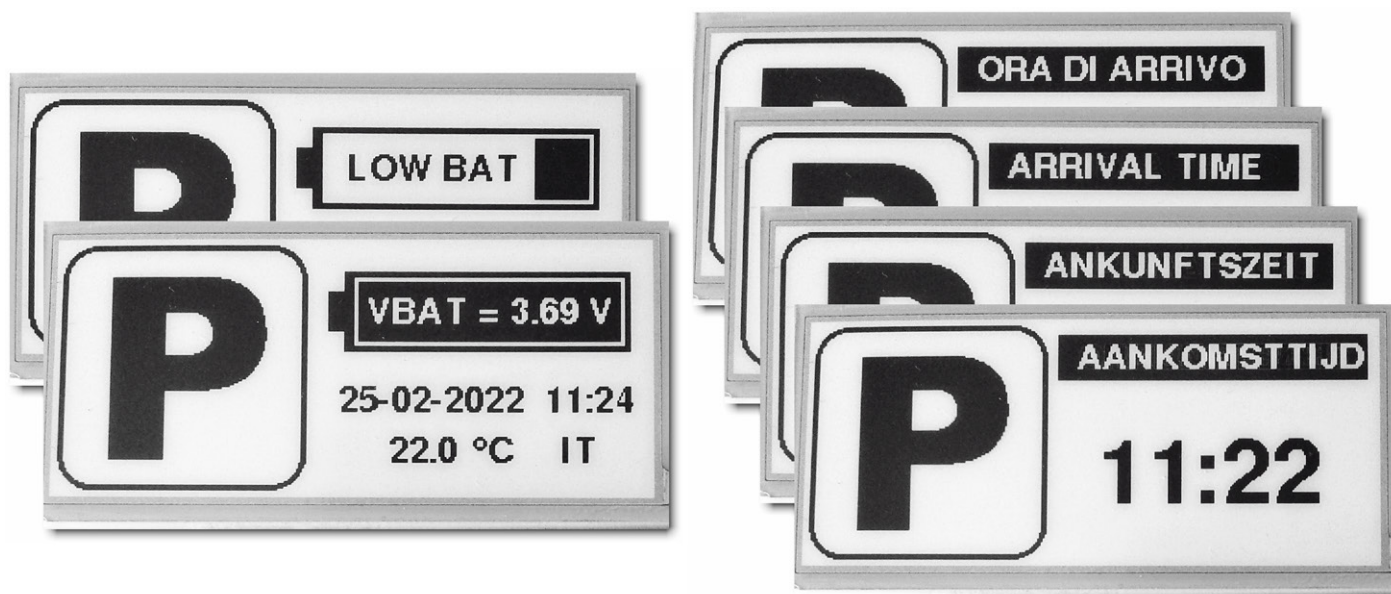


Figuur 1. Het werkingsprincipe het e-papier display (1: bovenste laag; 2: transparante elektrodel laag; 3: transparante microcapsules; 4: positief geladen wit pigment; 5: negatief geladen zwart pigment; 6: transparante olie; 7: pixel-elektrodel laag; 8: onderste laag; 9: licht; 10: wit; 11: zwart) (bron: <http://en.wikipedia.org/wiki/E-Ink>).

De evolutie van de automobielsector, vanuit het oogpunt van de integratie van elektronische technologie in voertuigen, heeft een zeer hoog niveau bereikt. Vrijwel elk functioneel aspect wordt optimaal beheerd door geavanceerde sensoren, digitale interfaces, microprocessoren en bijbehorende software. In de cockpit van wat nu kan worden beschouwd als 'een computer op wielen' is echter nog maar al te vaak een onmisbaar instrument te vinden, gemaakt van eenvoudig karton en handmatig bediend. Dit is de parkeerschijf, die nodig is om het begin van het parkeren in betreffend gereguleerde zones aan te geven. In de loop der decennia is dit accessoire vrijwel onveranderd gebleven – gemaakt van karton, plastic of andere meer edele materialen – en pas sinds kort zijn er digitale modellen op de markt. Het in dit artikel voorgestelde exemplaar maakt gebruik van een modern e-papier display met enkele speciale functies, zoals het instellen van de aankomsttijd met een enkele knop, een bericht in een keuze uit vier talen en indien gewenst de weergave van de huidige tijd en datum, de omgevingstemperatuur en het batterijniveau.

Het e-papier display

Deze relatief recente uitvinding (1996), de e-ink technologie (elektroforetische inkt), meestal e-papier genoemd [1], dankt zijn succes voornamelijk aan het gebruik ervan in e-boek readers of kortweg e-readers, de draagbare apparaten die een elektronisch alternatief bieden voor traditionele boeken, dankzij de papierachtige leeservaring en perfecte zichtbaarheid, zelfs bij veel licht. Het unieke kenmerk dat heeft geleid tot de verspreiding van deze technologie op andere gebieden is echter het vermogen om informatie gedurende lange tijd lange tijd weer te geven, zelfs zonder stroomvoorziening, waardoor apparaten kunnen worden gerealiseerd die in theorie alleen stroom nodig hebben gedurende de tijd die nodig is om het scherm bij te werken (refresh). Typische toepassingen, die steeds populairder worden in de detailhandel, zijn elektronische etiketten en prijskaartjes, vaak moeilijk te onderscheiden van papieren etiketten, die zo nodig kunnen worden bijgewerkt, zelfs op afstand met behulp van draadloze technologieën. Voor een beter begrip hoe elektronische inkt werkt, werpen we een blik op **figuur 1**.



Figuur 3. Voorbeeld van de info- en parking-schermen.

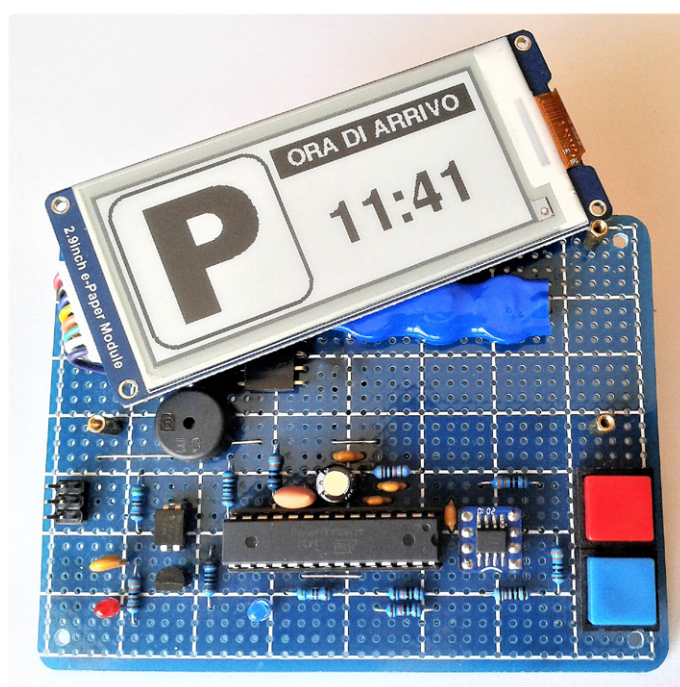
Hij communiceert met de microcontroller via een I²C-interface en een speciale bibliotheek. De knop SW2 (blauw in het prototype) 'wekt' de MCU, die zo gegevens over de huidige tijd kan ontvangen en deze op het display kan weergeven, samen met het bericht 'TIME OF ARRIVAL' en het 'P'-logo, alvorens terug te keren naar de slaapstand. Een korte pieptoon van de zoemer bevestigt het indrukken van de knop, terwijl de blauwe LED D2 blijft branden voor de duur van de gebeurtenis. SW1 (rood in het prototype) zorgt daarentegen voor een hardware-reset van de ATmega328P, onmiddellijk gevolgd door de weergave van een scherm met het 'P'-logo, het batterijsymbool en de batterijspanning, de omgevingstemperatuur, de huidige tijd en datum, en de geselecteerde taal voor het bericht, als in het voorbeeld van **figuur 3**.

Hetzelfde scherm wordt elke 24 uur automatisch weergegeven (op een door de gebruiker gewenst tijdstip, dankzij de RTC-alarm-functie) om het display compleet te verversen, zoals aanbevolen door de fabrikant, en om de batterijstatus en de goede werking van het hele systeem te controleren. Daarom kunnen we SW2 definiëren als 'parkeerknop' en SW1 als 'infoknop'. Weerstand R1 en R10 en de condensatoren C1 en C5 helpen eventuele naringheid als gevolg van de knop-contactdender te onderdrukken, terwijl R2, R4, R5 en R6 pull-up weerstanden zijn. C2, C3 en C4 zijn de gebruikelijke ontkoppelcondensatoren voor de voeding van de IC's. R11 beperkt de stroom die door de reset-aansluiting van het gebruikte display wordt getrokken. JP2 is de connector voor het display, terwijl JP1 het uploaden van firmware mogelijk maakt door een USBasp-programmer aan te sluiten. Vanwege het geringe stroomverbruik wordt de voeding van het prototype verzorgd door een kleine oplaadbare NiMH-accu van 3,6 V met een capaciteit van 40 mAh. De schakeling rond Q1, Q2, R8 en R9 is een constante-stroomregelaar met lage spanningsval, die de laadstroom beperkt tot ongeveer 6 mA, een waarde die dit type batterij zonder enig ongemak kan verdragen, zelfs als de aanbevolen laadduur van 14...16 uur wordt overschreden. Een micro-USB aansluiting maakt de aansluiting van een gewone 5V-lader

mogelijk via een zelfherstellende zekering ter beveiliging, terwijl D1 diode een 'backflow' vanuit de accu voorkomt. LED D3 geeft aan dat het apparaat aan het opladen is. Op het thema voeding komen we later nog terug.

Praktische uitvoering

Het prototype van de parkeerschijf werd zoals gebruikelijk opgebouwd op een experimenteerprint (**figuur 4**). Natuurlijk is dit slechts een van de vele mogelijkheden; met SMD-componenten



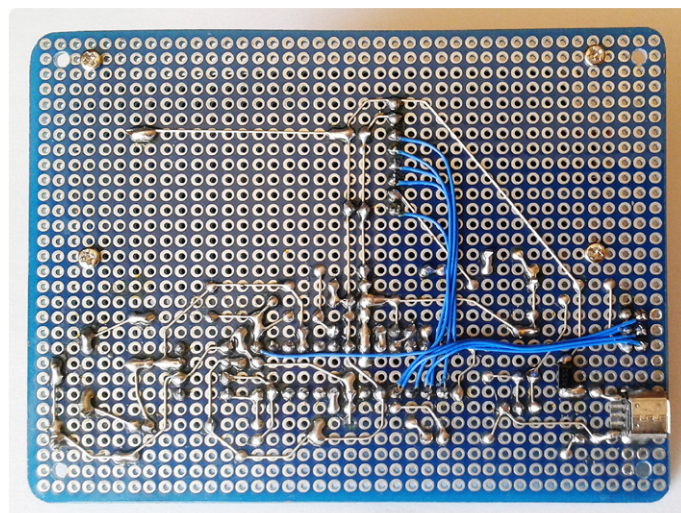
Figuur 4. De onderdelenzijde van het prototype.



Figuur 5. De in het project gebruikte e-papier module met zijn 'op maat gemaakte' connector (bron: Waveshare).

en een 'echte' print kunnen de afmetingen kleiner blijven – niet veel groter dan het display zelf. De accu en de zoemer bevinden zich onder de e-papier module, terwijl de achtpolige connector daarvan is samengesteld uit male en female headers/busstrips. De IC's zijn in voetjes gemonteerd, en voor de DS3231M is een gemakkelijk verkrijgbare SO-8 adapter gebruikt. De keuze van de behuizing is helemaal vrij – zolang deze een transparant venster heeft! Wat de e-papier module betreft, is het, zoals we al opmerkten, best wel moeilijk om niet te verdwalen tussen het zeer grote aanbod van schijnbaar identieke componenten. Het in het prototype gebruikte exemplaar is te zien in **figuur 5**. Het is een Waveshare-exemplaar met 2,9 inch diagonaal, zwart/wit, resolutie 296 x 128 [4]. Dit model heeft een ingebouwde level shifter en ondersteunt gedeeltelijke refresh, een onmisbare functie in dit project. Andere modellen maken wellicht niet dezelfde resultaten mogelijk die ik behaalde, of vereisen eventueel wijzigingen in de firmware of de bedrading. Ook moet rekening worden gehouden met het toegestane temperatuurbereik, in dit geval 0...50 °C.

Figuur 6 toont de soldeerzijde van het prototype, waar ook diode D1 (een SMD-type) en de micro-USB laadconnector zijn gemonteerd. Terug naar de voeding. De keuze voor een oplaadbare NiMH-accu, niet de meest recente technologie, is vooral ingegeven door het



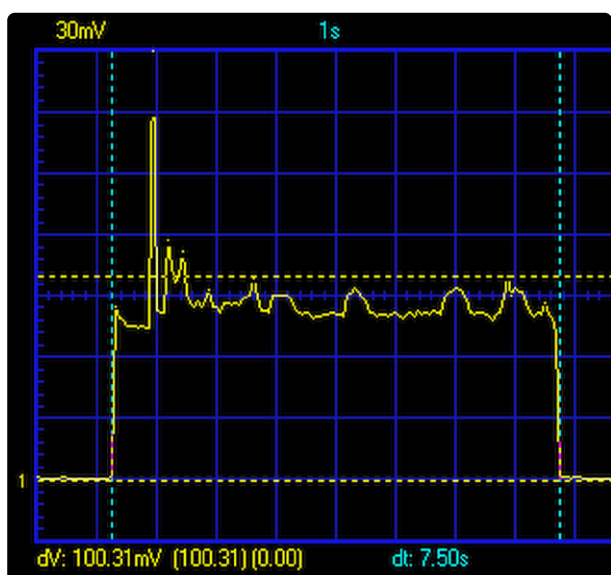
Figuur 6. De soldeerzijde van het prototype.

streven naar meer veiligheid en betrouwbaarheid in de niet altijd optimale gebruiksomstandigheden van een voertuig. In vergelijking met bijvoorbeeld Li-Ion accu's is het (uiterst gering) gevaar van ontploffing nihil en zijn de prestaties bij lage temperaturen zelfs beter. Normaal opladen duurt weliswaar langer, maar is gemakkelijker te realiseren. Gezien het minimale stroomverbruik en het feit dat voedingsspanning van de schakeling tussen 3,3 V en 5,0 V mag liggen, zou in ieder geval ook een oplossing met drie in serie geschakelde AAA-alkalinebatterijen kunnen worden overwogen, waardoor het laadcircuit overbodig wordt en men toch kan rekenen op een gebruiksduur die waarschijnlijk langer is dan die van de auto zelf! Het is niet eenvoudig om het 'verbruik' van de steeds populairder wordende batterijgevoede apparaten die afwisselend actief zijn en in de energiebesparende modus (ook wel slaapstand, power-down of spaarstand genoemd) verkeren, nauwkeurig te berekenen; helaas is het niet voldoende om een multimeter in serie met de voeding te plaatsen en de stroom te meten. De basisberekening die moet worden uitgevoerd om de gemiddelde waarde van de opgenomen stroom en dus een schatting van de levensduur van de batterij te verkrijgen, kan worden weergegeven met de volgende pseudo-formule:

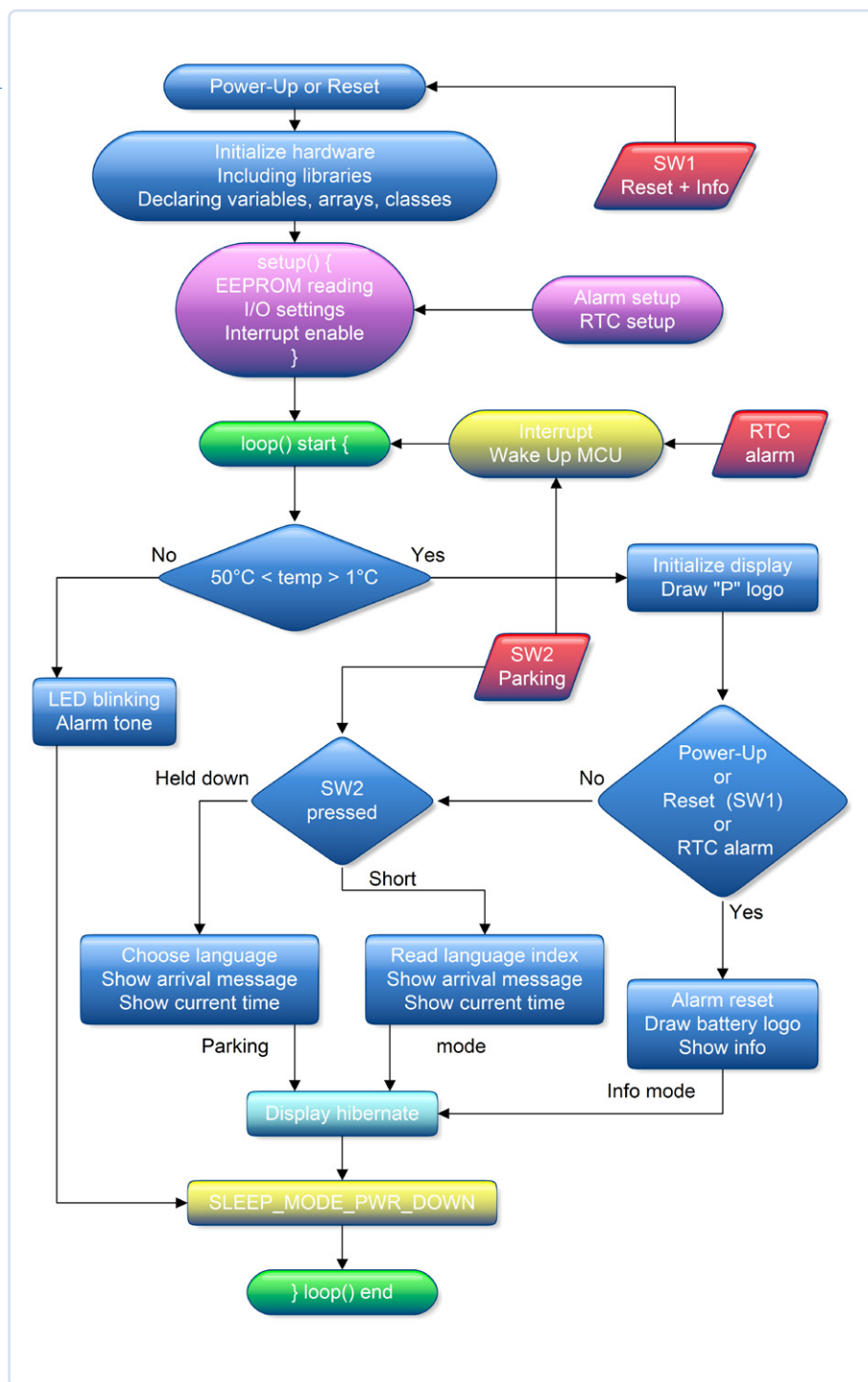
$$I_{AVG} = \frac{[(I_{ON} \cdot T_{ON}) + (I_{SBY} \cdot T_{SBY})]}{(T_{ON} + T_{SBY})}$$

waarbij I_{AVG} de gemiddelde stroom is, I_{ON} de stroom in actieve toestand, I_{SBY} de standby-stroom, T_{ON} de tijd dat de schakeling actief is en T_{SBY} de stand-by-tijd.

Ik heb het verloop van de stroom in actieve toestand 'gevangen' met de oscilloscoop (**figuur 7**) door de spanningsval over een kleine precisieweerstand in serie met de voeding te detecteren, resulterend in een aanvaardbare benadering van 10 mA gedurende 7,5 s, wat de duur van een weergavecyclus is. De ruststroom, gemeten met een digitale multimeter, bedraagt slechts 3 µA. Ervan uitgaande dat we de parkeerschijf vier keer per dag gebruiken, plus een



Figuur 7. Verloop van de door de schakeling opgenomen stroom.



Figuur 8: Stroomdiagram van de firmware.

kunnen voorzien (dus meer dan 200 dagen, minimaal in elk geval zes maanden)! Realistisch gezien zou zelfs de helft daarvan een goed resultaat zijn. Voor dit soort berekeningen zijn nu ook handige online-apps beschikbaar. Een link naar een app die ik als een van de meest effectieve en praktische heb ervaren is [5].

De firmware en zijn functionaliteit

De broncode voor dit project [9] is geschreven met de Arduino IDE 1.8.19 en vereist, voor een goede compilatie, de installatie van de Arduino-core *MiniCore* v2.1.3 en enkele specifieke bibliotheken. De gebruikte core staat een efficiënter en veelzijdiger beheer van de ATmega328P microcontroller toe en optimaliseert vooral het geheugengebruik van de gecompileerde code, die 31.264 van de 32.768 bytes programmeergeheugen (flash) en 1.501 van de 2.048 bytes dynamisch geheugen (SRAM) in beslag neemt, bijna aan de grens van de mogelijkheden van deze MCU. Er zij op gewezen dat dit project niet uitvoerbaar is, hoezeer je je best ook doet, met een Arduino Uno-board, aangezien een deel van het geheugen van dat bord gebruikt wordt door de bootloader om directe programmering mogelijk te maken, terwijl we voor de 'kale' microcontroller een externe USBasp-programmer gebruiken. En over geheugen gesproken: het sleutelwoord **PROGMEM** komt verschillende keren voor in de listing, verwijzend naar de read-only byte-arrays van de bitmap- en tekst-tekenreeksen. Door deze arrays als **PROGMEM** te declareren,

complete verversingscyclus, krijgen we een totale bedrijfsduur van $7,5 \times 5 = 37,5$ s en een rusttijd van $(24 \times 3600) - 37,5 = 86.362,5$ s gedurende 24 uur. Dat geeft:

$$I_{AVG} \approx \frac{[(10 \text{ mA} \cdot 3,75 \text{ s}) + (0,003 \text{ mA} \cdot 86362,5 \text{ s})]}{(37,5 \text{ s} + 86362,5 \text{ s})}$$

$$\approx 0,008 \text{ mA}$$

Een batterij van 40 mAh – we laten voor het gemak de afname van de capaciteit door zelfontlading, de afname van de nominale spanning, temperatuurschommelingen en andere factoren buiten beschouwing – zou bij een gemiddelde stroom van 0,008 mA de schakeling idealiter ongeveer $40 / 0,008 = 5.000$ uur van stroom

kunnen functies deze gegevens rechtstreeks uit het flash-geheugen lezen zonder ze eerst naar het veel kleinere SRAM te kopiëren; daar blijft dan ruimte over voor 'dynamische' uitvoering van het programma. We gebruiken de bibliotheek *DS3231M* 1.0.6 voor de communicatie met de geïntegreerde real-time klok, terwijl de bibliotheek *GxEPD2* 1.3.6, ondersteund door de grafische bibliotheek *GFX_Root* 2.0.0, is gekozen voor het basisbeheer van het e-papier display.

Deze laatste moet worden overschreven met de bij het project geleverde versie met gewijzigde fonts. De *GxEPD2*-bibliotheek is een enorm stuk werk, maar helaas ontbreekt gestructureerde documentatie; die is te vinden in de code van de talloze beschikbare voorbeelden die op het eerste gezicht afschrikwekkend complex lijken. Dat komt door de poging om de voorbeelden met zoveel mogelijk verschillende displaymodellen compatibel te maken.



Ik heb daarom geprobeerd een samenvatting te maken, waarbij ik alleen de functies en definities heb laten staan die nodig zijn voor het type display dat in het project wordt gebruikt. Deze zijn te vinden in het bestand *Waveshare_29_BW_avr.h*, terwijl het bestand *ParkBitmap128x128.h* de byte-array (verkregen door middel van een speciale converter [6]) bevat die de bitmapafbeelding van het parkeerlogo voorstelt (hoofdletter P in een vierkant met afgeronde hoeken, formaat 128 x 128 pixels, zwart/wit). Deze bestanden, die kunnen worden gedownload, bevinden zich in de sketch-map, samen met het hoofdbestand van de broncode *Disco_Orario_e-Paper.ino*, waarin zich ook links bevinden naar de kern- en bibliotheek-sites, uitgebreid commentaar op de code, en andere aanwijzingen die ik nuttig vond. Ik raad lezers die geïnteresseerd zijn in de details van de listing aan om deze te bestuderen door hem te openen met de Arduino IDE (of een andere favoriete editor). In plaats daarvan wil ik hier de werking van het programma op een meer beschrijvende manier illustreren, met behulp van het stroomdiagram van **figuur 8**.

Bedenkend dat de schakeling altijd is aangesloten op de batterij, worden bij de eerste keer inschakelen (*Power-Up*) de initialisatiebewerkingen en de `setup()`-functie uitgevoerd, met de activering van een interrupt, en vervolgens wordt `loop()` uitgevoerd.

Aan het begin van de loop wordt de omgevingstemperatuur gecontroleerd. Als deze niet binnen het acceptabele bereik ligt, wordt een hoor- en zichtbaar alarmsignaal gegeven, waarna de microcontroller in de maximale energiespaarmodus wordt geplaatst (*SLEEP_MODE_PWR_DOWN*). Anders gaat `loop()` verder met het initialiseren van het display, het weergeven van het 'P'-logo, de batterijstatus, de huidige datum en tijd, de temperatuur en de geselecteerde taal voor het aankomstbericht, met wat we *info*-modus kunnen noemen, waarna het display in 'slaapstand' wordt gezet (om energie te sparen), de MCU wordt uitgeschakeld en `loop()` wordt afgebroken. Vanuit deze toestand kan de microcontroller worden 'gewekt' (*Wake Up*) door een hardware-reset (met de SW1-knop) of door een interrupt, een gebeurtenis die in dit geval wordt gegenereerd door de dagelijkse alarmfunctie van de RTC of door de knop SW2. Als de hervatting is veroorzaakt door de reset of het alarm, begint `loop()` opnieuw vanaf het begin en eindigt hij altijd in de *info*-modus. Maar als de microcontroller is gereactiveerd via SW2, start `loop()` opnieuw, initialiseert het display, toont het logo en als de knop werd ingedrukt en onmiddellijk losgelaten, toont hij het aankomstbericht en de huidige tijd, in wat we de *parking*-modus zullen noemen. Door SW2 daarentegen ingedrukt te houden, worden de berichten in de vier talen na elkaar weergegeven. Laat de knop los wanneer de gewenste taal verschijnt; die instelling wordt dan opgeslagen in EEPROM tot de volgende wijziging. De huidige tijd verschijnt dan, en de cyclus eindigt altijd in de *parking*-modus. Een voorbeeld van de verschillende schermen is te zien in figuur 3. We hebben gezien hoe de huidige datum en tijd worden geleverd en onderhouden door de geïntegreerde RTC DS3231M, die nog moet worden geprogrammeerd na de eerste maal inschakelen van de voedingsspanning

(of wanneer de voedingsspanning is uitgevallen). Om de firmware en de schakeling te vereenvoudigen en de toevoeging van verdere knoppen te vermijden, worden de datum en de tijd gelijktijdig met het laden van de sketch geprogrammeerd via een speciale regelcode die in de `setup()`-routine is ingevoegd:

```
DS3231M.adjust(DateTime(2022, 03, 02, 19, 10, 00));
```



Onderdelenlijst

Weerstanden (0,25 W, 1%):

R1,R10,R12 = 330 Ω
R2,R4,R5,R6 = 10 k
R3,R8 = 5k6
R7 = 3k3
R9 = 100 Ω
R11 = 1 k

Condensatoren:

C1,C3,C5 = 100 n/50 V polyester of multilayer keramisch
C2 = 100 μ /10 V elektrolytisch
C4 = 470 n/50 V polyester of multilayer keramisch

Halfgeleiders:

Q1 = IRLD024 logic-level gate drive MOSFET
Q2 = BC546B
D1 = SS34 low-drop Schottky-diode
D2 = LED 3 mm, blauw, Hi-Bright
D3 = LED 3 mm, rood, Hi-Bright
U1 = ATmega328P-PU
U2 = DS3231M RTC (SO-8)

Diversen:

Y1 = 8 MHz keramische resonator
BUZZER = piëzo-elektrische zoemer
SW1,SW2 = druktoets maakcontact
F1 = zelfherstellende zekering (polyfuse) 250 mA
X1 = micro-USB Socket type B
JP1 = 6-pins ICSP printconnector, male
JP2 = 8-pins printconnector male/female
DISPLAY = e-Paper display 2,9 inch B/W (zie tekst)
BAT1 = NiMH 3,6V accu, 40 mAh (3 x V40H)
28-polige DIP-voet
8-polige DIPvoet
SMD - DIP8/SO-8 adapter
experimenteerprint (gaatjesprint)



Het te gebruiken formaat is van het type “JJJJ,MM,DD,hh,mm,ss”. Zodra je de juiste gegevens hebt ingevoerd, kun je de sketch laden, de datum en tijd controleren, de regel uitcommentariëren (voeg de dubbele slash toe aan het begin) en de sketch opnieuw laden (dit is om te voorkomen dat de klok bij elke reset terugkeert naar zijn oorspronkelijke instelling). Om voldoende nauwkeurigheid te bereiken, meet je gewoon de tijd die nodig is om de code te uploaden, zeg 30 seconden, en begin je de eerste upload 30 seconden voor de geprogrammeerde tijd. Met wat oefening kan een synchronisatie op de seconde nauwkeurig worden bereikt... zonder extra kosten! De dagelijkse verversingstijd wordt ook in de code geprogrammeerd, gewoon door de gewenste tijd in te voeren. Tot slot een opmerking over de methode die wordt gebruikt om de batterijspanning te meten. Die is in verschillende varianten te vinden op het Internet, en ook Microchip zelf noemt hem in een eigen application note [7]. De ‘truc’ is om via de juiste registers de interne referentiespanning (1,1 V) in te stellen alsingangswaarde van de ADC en de te meten spanning (VCC) als referentie. Elke verandering in VCC zal de ADC-uitlezing veranderen, zodat de spanning met voldoende nauwkeurigheid kan worden berekend.

Afsluitende overwegingen

Hoewel de e-papier technologie ontegenzeggelijk interessante kenmerken heeft, die hem bijzonder geschikt maken voor het onderhavige project, heeft hij ook bepaalde beperkingen, waarvan de meest in het oog springende de lage verversingsfrequentie is. Het hier gebruikte model voert de volledige verversingscyclus uit in 2 seconden en de gedeeltelijke verversing in 0,3 seconden. Deze waarden kunnen dus niet concurreren met andere beeldschermtypes bij de visualisatie van snel veranderende beelden, grafieken en tekst. Tenslotte hoop ik, zoals ik pleeg te zeggen, dat je naast de eigenlijke bedoeling van het voorgestelde project een aantal interessante ideeën hebt gekregen om uit te werken en te hergebruiken – met de mentaliteit van een Maker – en dat dit artikel je zin en nieuwsgierigheid heeft gewekt om te experimenteren met e-papier displays! ◀

230012-03

Vragen of opmerkingen?

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



Over de auteur

Antonello Della Pia voelde zich van jongs af aan aangetrokken tot elektriciteit en elektronische apparaten. Hij is in het bezit van het diploma ‘Technicus elektrotechniek’ van de middelbare beroepsopleiding. Antonello heeft altijd zijn passie voor analoge en digitale elektronica gecultiveerd en ontwikkeld. Momenteel speelt hij met microcontrollers en programmeren en probeert hij zijn computervaardigheden te verbeteren. Antonello houdt ervan projecten te ontwikkelen en voor te stellen die zo origineel mogelijk en – naar hij hoopt – ook interessant zijn.



Gerelateerde producten

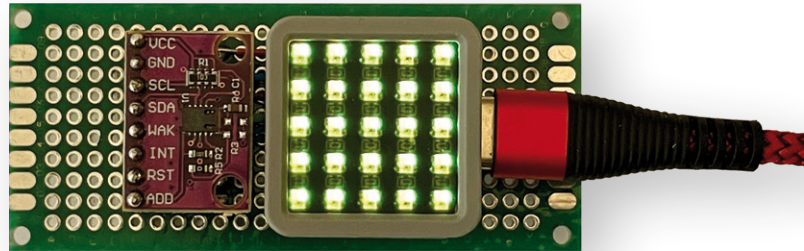
- ▶ **Waveshare 2.9" 3-color E-Ink/E-Paper Display Module (SKU 18776)**
www.elektor.nl/18776
- ▶ **Ynvisible Segment E-Paper Display Kit (SKU 20143)**
www.elektor.nl/20143



WEBLINKS

- [1] Elektronisch papier (Wikipedia): https://nl.wikipedia.org/wiki/Elektronisch_papier
- [2] Datasheet ATmega328P: <https://www.elektormagazine.com/atmega328p-datasheet>
- [3] Datasheet DS3231M: <https://datasheets.maximintegrated.com/en/ds/DS3231M.pdf>
- [4] Waveshare - 296x128, 2.9 inch E-Ink displaymodule: <https://www.elektormagazine.com/waveshareeink>
- [5] Online berekening levensduur batterij: <https://www.omnicalculator.com/other/battery-life>
- [6] BitmapToByteArrayConverter: <https://www.briandorey.com/post/bitmap-byte-converter-for-e-ink-display>
- [7] Measure VCC/Battery Voltage Without Using I/O Pin..., Microchip: <https://www.elektormagazine.com/vcc/battery>
- [8] Werkingsprincipe van het e-papier display: <https://www.elektormagazine.com/bwcapsules>
- [9] Broncode: <https://www.elektormagazine.nl/230012-03>

eCO₂ Telegram-bot



meet de luchtkwaliteit via Telegram

Peter Neufeld (Duitsland)

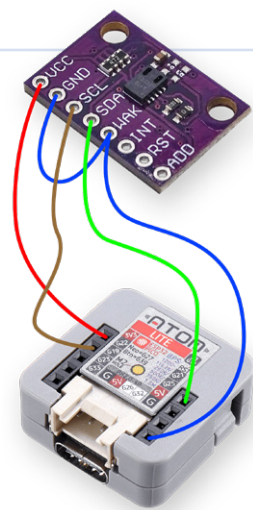
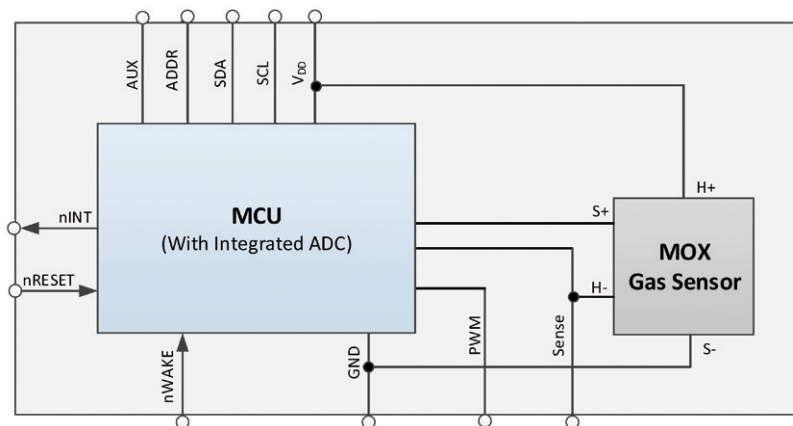
Zeer nauwkeurige en dure meetapparatuur is niet altijd nodig om de luchtkwaliteit binnenshuis te bepalen. Een ruwe indicatie en bewaking van grenswaarden kunnen ook zeer nuttig zijn als je gewoon wil weten wanneer het nodig is om een intensief gebruikte ruimte te luchten. Zo'n apparaat wordt nog nuttiger wanneer het eenvoudige lokale display wordt uitgebreid met een alarmmelding naar een Telegram-account op een smartphone.

Sterk geïntegreerde sensortechnologie, een moderne microcontroller en een beetje BASIC-programmering maken het mogelijk om met deze eCO₂-monitor en zijn verschillende weergavemodi op vier manieren aandacht te vestigen op een slechte luchtkwaliteit:

- lokale NeoPixel LED(s) als luchtkwaliteits-verkeerslicht;
- webinterface voor lokale apparaten met een webbrowser;
- handmatig opvragen via Telegram Messenger;
- Telegram-waarschuwing verzonden naar een specifieke Telegram-gebruiker.

De hardware

De eenvoudige schakeling in **figuur 1** is gebaseerd op een eCO₂-sensor CCS811 en een ESP32 SoC-module. De luchtkwaliteit wordt weergegeven door een of meer NeoPixel LED's. Een enkele drukknop is verbonden met een ingang. De sensor en de SoC communiceren via een tweedraads I²C-bus. Het NeoPixel-display gebruikt slechts een datalijn op GPIO27, of nu een enkele LED-dot of een matrix wordt geactiveerd.



Figuur 1. eCO₂ Telegram-bot met M5Stack ATOM matrix en CCS811 inclusief interne schakelingen (bron: AMS-datasheet).

Hoewel bijna elke ESP32-module met vrije I²C-pinnen geschikt is voor deze klus, raad ik een "ATOM matrix" of "ATOM lite" van M5Stack aan. Deze handige apparaatjes combineren namelijk een ESP32-PICO-D4 met een antenne, een NeoPixel-matrix of een enkele NeoPixel-dot, een drukknop en nog wat meer in een kleine en stijlvolle stevige behuizing. Veel meer hardware dan de sensor en de ESP32-module heb je dan niet meer nodig om het geheel op een stukje gaatjesprint op te bouwen. Daarvoor prik je de CCS811 en de ATOM in de juiste busstrips en headers en maak je de weinige noodzakelijke verbindingen met stukjes draad, zoals getoond in **figuur 2**. De onderkant van de print is geïsoleerd met plakband.

De schakeling wordt via de ESP32-module gevoed uit een USB-voeding (5 V bij maximaal 500 mA), naar keuze via de USB-C connector, de HY2.0 Grove-connector of de onderste busstrips.

De CCS811 eCO₂-Sensor

De CCS811-sensor kan het CO₂-gehalte niet rechtstreeks meten! Hij berekent het equivalente CO₂-gehalte (eCO₂) door de tVOC-waarde (Total Volatile Organic Compound) te bepalen, waarbij de belangrijkste bron van deze vluchtige organische stoffen de door mensen uitgeademde lucht is.

Deze goedkope metal oxide semiconductor (MOS) sensor zoekt zelf naar een (relatieve) basiswaarde (baseline) voor 'goede lucht' door de beste luchtconditie over een langere periode te bepalen en vervolgens aan te nemen dat de sensor zich op dat moment in frisse, onvervuilde lucht met 400 ppm CO₂ bevond. De sensor slaat deze waarde niet zelf op. Bovendien kan de gevoeligheid van de sensor in de loop der tijd en onder verschillende omgevingsomstandigheden zoals temperatuur en vochtigheid, veranderen. Voor betrouwbare eCO₂-metingen moet de sensor dus:

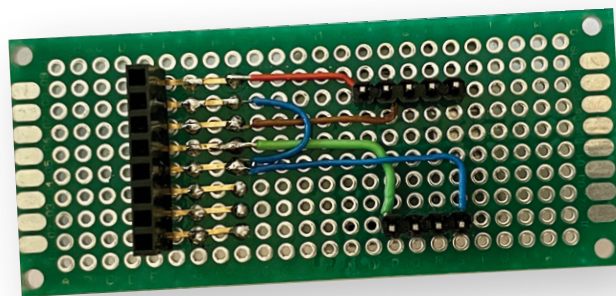
- aan een burn-in periode van meer dan 48 uur zijn onderworpen;
- een minimale aanloopperiode van ongeveer 20 minuten krijgen na elke koude start.

De datasheet [1] bevat meer gedetailleerde informatie over de gassensor.

De software

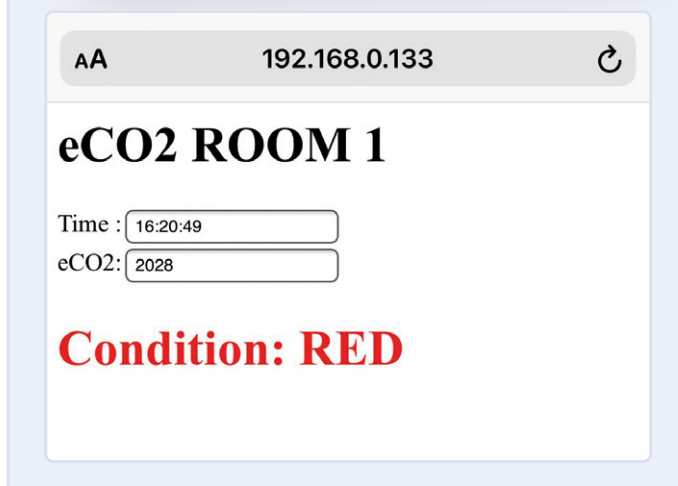
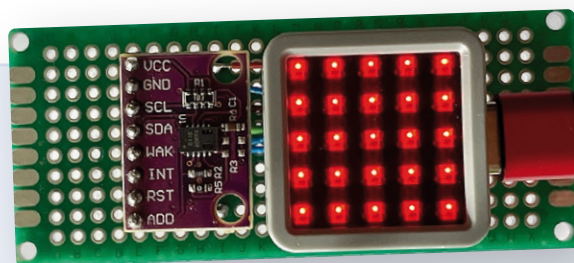
Het programma is ontwikkeld met Annex32, een BASIC-interpreter voor ESP32 [2]. Na het programmeren van de interpreter in de flash-RAM van de ESP32-module met behulp van een installatieprogramma en een seriële USB-interface, draait de interpreter samen met zijn ontwikkelomgeving volledig in de ESP32. Alleen een Chrome- of Firefox-browser is nodig om het BASIC-script te laden, te bewerken, te testen en (automatisch) uit te voeren. De minimaal vereiste Annex32-versie is V1.435, omdat deze ondersteuning biedt voor de CCS811 en Telegram messenger. De online helpfunctie [3] voor Annex32 biedt een zeer goede inleiding op deze BASIC-interpreter. De belangrijkste taken van de BASIC-code zijn:

- De CCS811 wordt geïnitieerd; eenmaal per seconde wordt de eCO₂-waarde opgevraagd.
- De eCO₂-status van de omgevingslucht wordt geklassificeerd als *groen, geel of rood*.



Figuur 2. De simpele bedrading van de twee modules op gaatjesprint.

- De ingebouwde NeoPixel-matrix of de enkele NeoPixel-LED geeft deze categorie lokaal aan door zijn kleur.
- Een webinterface toont de eCO₂-waarde en de classificatie in een browser via het (W)LAN.
- De classificatie en de eCO₂-waarde kunnen handmatig worden opgevraagd via Telegram messenger, want het programma werkt met een Telegram-bot en haalt inkomende gebruikersopdrachten op van de Telegram-server.
- Een Telegram-waarschuwing wordt automatisch naar het laatste Telegram-chat_id verstuurd wanneer de luchtkwaliteit **rood** is (**figuur 3**).
- De basiswaarde van de luchtkwaliteit kan handmatig worden opgeslagen door op de voorste knop van de ATOM-module te drukken (of via een inkomend Telegram-commando).

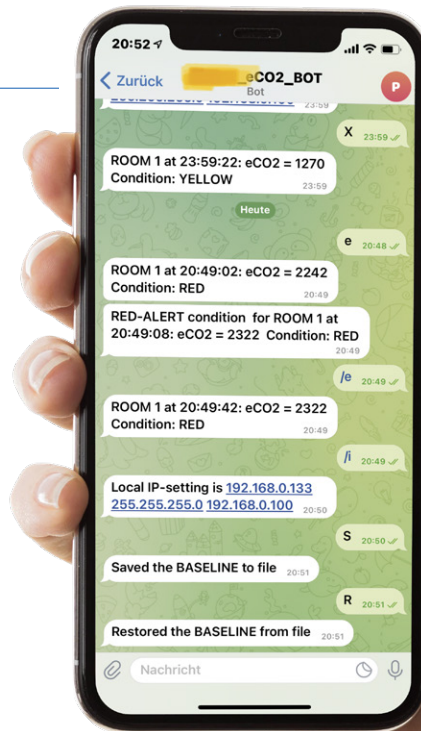


Figuur 3. Alarm – de lucht is echt slecht!



Programmafragment: webpagina-uitvoer

```
'#####
MAKE_WEBPAGE:
'-----
'create the autorefreshing html page with dynamic
'display values and matching colors
A$ = ""
A$ = A$ + "<H1>eCO2 " + LOCATION$
A$ = A$ + "</H1>"
A$ = A$ + "Time :" + textbox$(T$) + "<br>"
A$ = A$ + "eCO2: " + textbox$(STR_eCO2$) + "<br>"
A$ = A$ + |<span style="color:| + COND_COL$ + |">|
A$ = A$ + "<H1>Condition: "+ CONDITION$+ "</H1>"
A$ = A$ + "</span>"
cls
autorefresh 1000
html A$
return
'#####
```



Figuur 4. Uitlezing via de Telegram-bot.

De Telegram-bot routine vraagt regelmatig bij de Telegram-servers eventuele inkomende commando's op. Hij reageert op deze commando's zoals te zien in **figuur 4**.

- > **/e** levert de eCO₂-waarde en de klassificatie (*groen/geel/rood*).
- > **/s** slaat de basiswaarde op in */baseline.txt*.
- > **/r** herstelt de basiswaarde uit */baseline.txt*.
- > **/i** retourneert de lokale IP-instellingen van de module.
- > [elk ander karakter] doet hetzelfde als **/e**.

Je eigen Telegram-token

Om de Telegram-functies in het BASIC programma te gebruiken, moet je eerst je eigen Telegram-bot aanmaken door de instructies van BotFather te volgen onder [4] in je Telegram-app. Dan krijg je je persoonlijke Telegram-token en een botnaam. Belangrijk: je moet deze twee gegevens in het BASIC-programma invoegen om de overeenkomstige variabelen in te stellen.

Het gebruik van Annex32-BASIC, waarvan je de huidige versie altijd kunt ophalen in het Annex RDS Forum [5], heeft in ieder geval het voordeel dat het zelfs voor onervaren programmeurs gemakkelijk te lezen is en bovendien aan je eigen behoeften kan worden aangepast. Voor een beter begrip van de functies is de code [6] rijkelijk voorzien van commentaar. Dit script wordt vervolgens ingevoerd in de web-editor van Annex32 en opgeslagen in de ESP32-module als autorun-bestand (*/default.bas*). ◀

vertaling: Willem den Hollander — 210566-03

Vragen of opmerkingen?

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



Gerelateerde producten

- > **M5Stack AtomU ESP32 Development Kit (SKU 20184)**
www.elektor.nl/20184
- > **ESP32 PICO Kit V4**
met headers: www.elektor.nl/18423
zonder headers: www.elektor.nl/20323
- > **Dogan and Ahmet Ibrahim: The Official ESP32 Book**
www.elektor.nl/18330

WEBLINKS

- [1] CCS811 datasheet: <https://bit.ly/2qQKqKu>
- [2] AnnexToolKit 1.1 inclusief versie 1.48.22:
<https://bit.ly/3Gtf8RS>
- [3] Annex32 WIFI RDS Help Version 1.48.2:
<https://bit.ly/3Gwhdg9>
- [4] BotFather op Telegram: <https://t.me/BotFather>
- [5] Annex RDS-forum: <https://bit.ly/3VBTRK2>
- [6] Dit project op Elektor Labs:
<http://www.elektormagazine.com/labs/eco2-telegram>



Achter de schermen van DHZ high-end audio

Ton Giesberts van Elektor over de fijne kneepjes van het analoog ontwerpen

Jan Buiting (Elektor)

Maak kennis met de man achter de Elektor Fortissimo-100 eindversterker en vele andere high-end audioprojecten: Elektor Labs-medewerker Ton Giesberts.

Jan: Kun je ons in het kort iets vertellen over je technische achtergrond, je ervaring en hoe en wanneer je bij Elektor bent komen werken?

Ton: Na mijn studie werd ik vrijwel direct benaderd door een uitzendbureau "om eens langs te komen voor een gesprek". Ze boden me verschillende banen aan, waaronder eentje op een school. Dat sprak me niet aan, maar wel een baan bij een uitgever van een technisch maandblad, gevestigd in Beek. Als abonnee wist ik dat dit alleen maar *Elektuur* kon zijn. Ik had al verschillende schakelingen uit het blad gebouwd, en na een sollicitatiegesprek heb ik eerst een paar maanden via het uitzendbureau in het lab gewerkt voordat ik vast in dienst kwam. *Elektuur* was juist op zoek naar iemand die analoge techniek wilde doen, dus dat kwam me prima uit.

Jan: Wat zijn – buiten je beroep – je favoriete bezigheden of hobby's?

Ton: Elektronica was en is nog steeds mijn tijdverdrijf. Ik ontwerp ook wat dingen buiten het werk om. Ik lees graag en kijk naar documentaires of een goede film.

Jan: Hoe heb je in de loop der jaren het 'elektronica-ambacht' en de praktijk ervan zien veranderen?

Ton: In de loop der tijd is het gebruik van through-hole componenten afgenomen. Ik veronderstel dat de opkomst en het standaardgebruik van meestal SMD-onderdelen het thuis solderen steeds minder

aantrekkelijk heeft gemaakt. Elektronica is in hoog tempo verschoven naar software-ontwikkeling. Microprocessor-kaarten zijn al lang populair, maar de zelfbouw ervan is minder aantrekkelijk dan het kopen van kant-en-klare modules, vooral de laatste jaren. De software bepaalt dan de toepassing, zij het uiteraard met wat externe (analoge) componenten zoals sensoren en dergelijke.

Jan: Hoe is je werk tegenwoordig georganiseerd, en met welk instrumentarium en gereedschap werk je meestal? Gebruik je een simulatieprogramma voor je ontwerpen?

Ton: Nu ik voor een andere afdeling binnen Elektor werk [*boeken, kits en het kernassortiment – noot van de redactie*], zijn er andere zaken waar ik me mee bezig moet houden. Dat betreft dan vaak niet het ontwikkelen van nieuwe schakelingen of het verwerken van ruw materiaal dat ons rechtstreeks wordt aangeleverd, maar veeleer het controleren en bestellen van componenten, het ondersteunen van collega's en nog veel meer. Zoals de meeste elektronica-ingenieurs gebruik ik meestal een multimeter en een oscilloscoop. Natuurlijk zijn een paar labvoedingen en een functiegenerator onmisbaar, en niet te vergeten een audio-analyzer. Naast een gewone soldeerbout gebruik ik een reflow-oven en hete lucht om SMD's te solderen. Voor het fotograferen van prototypes heb ik een kleine opstelling op zolder om goed belichte foto's te maken. Al sinds de goede oude DOS-dagen gebruik ik bij voorkeur Micro-Cap voor simulaties.

Jan: We horen dat je een Audio Precision analyzer bezit. Hoe bevalt het werken met een instrument waar veel audiofielen alleen maar van kunnen dromen?

Ton: Zonder een goede analyzer is het ontwerpen van high-end audioschakelingen bijna onmogelijk. Natuurlijk, en gelukkig maar, zijn meetgegevens niet alles, maar ze zijn geweldig om onvolkomenheden aan

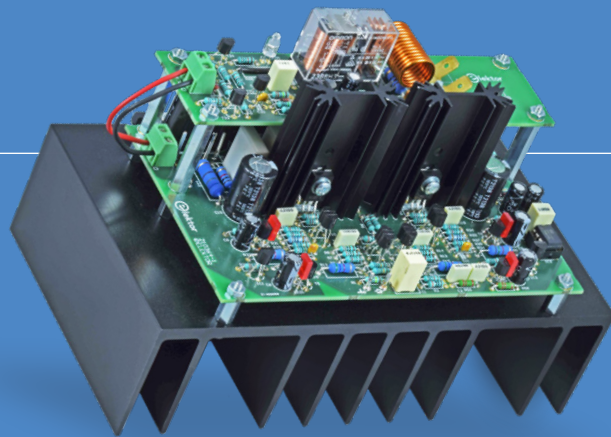
het licht te brengen. Met een oscilloscoop en een multimeter kom je niet verder. Ik ben echt tevreden dat ik de AP-analyzer al vele jaren kan gebruiken. Maar waar het om gaat is hoe en wat je meet. Met andere woorden: weet wat je meet! Zonder de nodige ervaring om de juiste conclusies te trekken, kun je met die AP weinig bereiken. Dat leer je niet in één dag.

Jan: Wat zijn je belangrijkste inspiratiebronnen voor nieuwe ontwerpen, onderdelen en technieken?

Ton: Voor nieuwe ontwerpen zijn de oudjes een zeer goede inspiratiebron. Op internet zijn er genoeg te vinden. De ontwikkeling van bipolaire vermogenstransistoren heeft bijvoorbeeld niet stilgestaan. Wat is er dus mis mee om een oud ontwerp weer op te pakken, het hier en daar aan te passen, en moderne componenten te gebruiken om het nieuw leven in te blazen?

Jan: Je werk lijkt vooral gericht op analoge elektronica, eventueel gecombineerd met wat digitaal spul, maar dat blijft dan beperkt 'periferie'. Hoe weeg je de relatieve betekenis van analoog en digitaal binnen je ontwerpen?

Ton: Ik blijf een liefhebber van de doodgewone standaard logische families, ook als drivers of in combinatie met analoge schakelingen. Maar ik apprecieer inmiddels dat microcontrollers je ontwerp hardwarematig goedkoper en kleiner kunnen maken, terwijl software een project veelzijdiger maakt. Ik heb programmeren nooit echt leuk gevonden; analoge technologie lijkt tastbaarder, maar dat is misschien maar een idee. Als ik al een microcontroller gebruik, is het er een uit de AVR-familie, en dan is programmeren met BASCOM zelfs voor mij nog te doen. Af en toe is het zelfs een leuke en welkome afwisseling. Tegenwoordig kan het nog eenvoudiger met tools als CircuitPython en een Raspberry Pi Pico.



Jan: Je heeft bijgedragen aan de lange lijst van high-end audioversterkers die Elektor beroemd hebben gemaakt. Welke waren het beroemdste en welke is je persoonlijke favoriet?

Ton: Een van de eerste waaraan ik werkte was de LFA150 [1] (hoewel niet mijn ontwerp) en zijn afgeleide, de LFA-50-OA [2]; daarna kwam de Power Amp [3], en dan later de Audio Drive, een versterker met bescheiden vermogen. De HEXFET power amp [4] plus diens opvolger, de IGBT power-amp [5] (met dezelfde print) met speciaal voor audio ontwikkelde IGBT's, alsmede eindversterkers voor actieve subwoofers en diverse actieve luidsprekersystemen. Later kwamen nog de Crescendo Millennium Edition [6] en de ClariTy eindversterker [7]. Meer recentelijk een 200W klasse-D audio-eindversterker [8] en de Compacte audio-eindtrap [9], en natuurlijk de Fortissimo-100 [10] en nog veel meer grote en kleine versterkers. De Power Amp werd destijds getest en goed bevonden door een Duits audiotijdschrift, en dat is misschien de reden waarom ik er de voorkeur aan geef, maar ik wil niet echt een favoriet kiezen.

Jan: Je Fortissimo-100 versterker werd relatief geruisloos ontwikkeld via de Elektor Labs-website, kreeg een grandioze Jumpstart en werd nog beter verkocht na de 'officiële' publicatie in Elektor. Kun je iets vertellen over het ontwerp- en publicatieproces. Wat waren de struikelblokken?

Ton: Mij werd gevraagd weer een nieuwe eindversterker te ontwerpen. Ik dacht daarbij vooral aan de Power Amp en heb dat concept ver gevolgd, afgezien van de eindtrap. Het probleem was (en is) dat grote dubbeltransistoren zoals de MAT02 en MAT03 niet meer geproduceerd worden, en dat de kleine

versies in SMD niet zo goed presteren. Ik ben daarom teruggekeerd naar de goede oude BC546B en BC556B, waarmee een zeer snelle versterker kan worden gemaakt bij een hogere voedingsspanning. Natuurlijk geeft men nu de voorkeur aan Miller-compensatie, maar ik wilde het oorspronkelijke concept zo min mogelijk veranderen. Ik gebruikte alleen ThermalTrack-eindtransistoren voor de eindtrap, waardoor ik een symmetrische bootstrap kon proberen – aanvankelijk als proof-of-concept. Dit werkte verrassend goed en bracht een grotere uitgangstrap binnen bereik. Voor een betere selectie van de BC-torren in de kit werden ze op zoveel mogelijk plaatsen gebruikt, inclusief de stroombronnen van de verschilversterkers. Om het vermogen in toom te houden zijn collectorweerstand nodig, die op hun beurt het gebruik van onder andere een gestabiliseerde voedingsspanning onontkoombaar maken.

Jan: We lezen op Elektor Labs dat er een lineaire voeding in de maak is, speciaal voor de Fortissimo-100. Waarom? Die SMPS800RE schakelende voeding voldoet toch prima?

Ton: Voor sommigen is het gebruik van een schakelende voeding vloeken in de kerk en een absolute 'no go'. Maar er bestaan tegenwoordig goede exemplaren, speciaal voor audio ontworpen. Een daarvan is de SMPS800RE. Die maakt ook het hele voedingsdeel lekker compact. Toegegeven,

het ruisniveau van een schakelende voeding is hoger en er is wat RF-rimpel. Maar die ligt buiten het hoorbare bereik en wordt door de versterker effectief onderdrukt. Om toch aan de wensen van audiofiele geïnteresseerden tegemoet te komen, heb ik overwogen een analoge regelaar te bouwen. Maar die moest klein blijven en gemakkelijk na te bouwen. Helaas is ontwerpen heden ten dage een hele uitdaging door de slechte, late of zelfs 'on' beschikbaarheid van componenten. Al snel ben je gedwongen om compromissen te sluiten; daarom wordt in dit ontwerp gebruik gemaakt van 'ouwetjes' zoals TIP35C en TIP36C – nog steeds gemaakt door diverse fabrikanten en uit voorraad leverbaar bij diverse distributeurs.

Jan: Hoe en waar zie je jezelf over een jaar aan werken?

Ton: Ik verwacht min of meer hetzelfde te doen als nu, maar het gaat momenteel niet goed met de wereld, zodat het voorspellen van de toekomst gissen blijft. ◀

220603-03



Gerelateerde producten

➤ **Elektor Fortissimo-100 Power Amplifier Kit (SKU 20273)**
<https://elektor.nl/20273>

WEBLINKS

- [1] "LFA150 – een snelle eindversterker", Elektor november 1988: <http://www.elektormagazine.nl/magazine/elektor-198811/45754>
- [2] "LFA-50-OA", Elektor oktober 1991: <http://www.elektormagazine.nl/magazine/elektor-199110/37655>
- [3] "The power amp", Elektor oktober 1990: <http://www.elektormagazine.nl/magazine/elektor-199010/37422>
- [4] "HEXFET power amp", Elektor november 1993: <http://www.elektormagazine.nl/magazine/elektor-199311/38135>
- [5] "IGBT power-amp", Elektor juni 1995: <http://www.elektormagazine.nl/magazine/elektor-199506/38437>
- [6] "Crescendo Millennium Edition", Elektor april 2001: <http://www.elektormagazine.nl/magazine/elektor-200104/13684>
- [7] "ClariTy eindversterker", Elektor juni 2004: <https://www.elektormagazine.nl/magazine/elektor-200406/14418>
- [8] "200W Class-D Audio Power Amplifier", Elektor Labs, 10 augustus 2016:
<https://elektormagazine.com/labs/200w-class-d-audio-power-amplifier-150511>
- [9] "Compacte audio-eindtrap", Elektor september 2013: <https://www.elektormagazine.nl/magazine/elektor-201309/23286>
- [10] "Fortissimo-100 high-end versterker", Elektor november 2022: <http://www.elektormagazine.nl/magazine/elektor-282/61137>

Oost West Lab Best

werk in uitvoering...

Figuur 1. Dit pingpongballetje balanceert in/op een waterstraal.

Ilse Joostens (België) en Eric Bogers (Elektor)

Voor de trouwe lezers van Elektor is Ilse Joostens geen onbekende – als auteur van de column “Uit het leven gegrepen”, maar ook als de hogepriesteres van vintage Nixie- en VFD-buisjes. De Corona-pandemie met zijn lockdowns en andere ellende heeft ook haar niet onberoerd gelaten, zoal uit het onderstaande verhaal moge blijken.

De ontstaansgeschiedenis van het apparaat dat in deze aflevering van “Oost West...” de hoofdrol speelt, gaat vele jaren terug. We laten Ilse aan het woord:

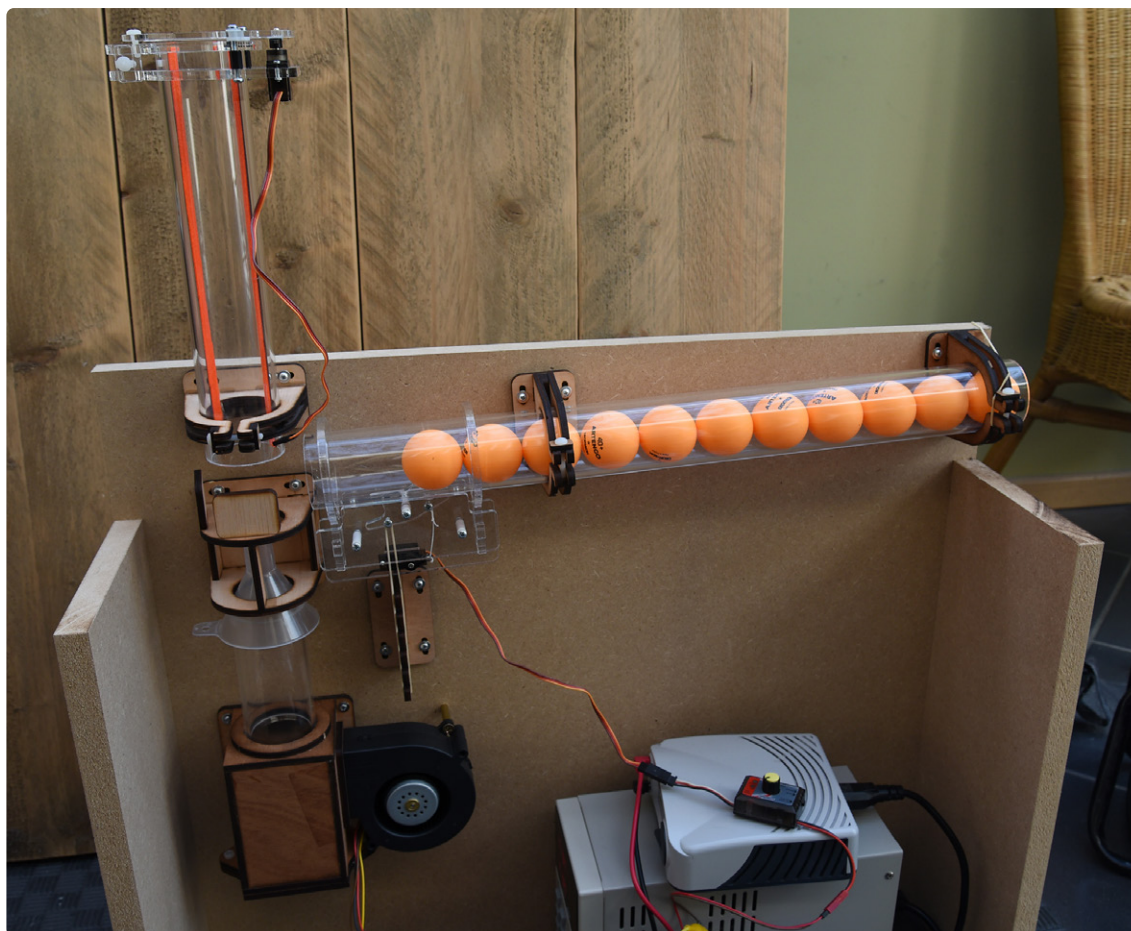
“Zo’n twintig jaar geleden woonde ik ‘vrijgezellig’ in Dendermonde en zocht aansluiting bij een vereniging om activiteiten te gaan ondernemen. Nu wilde het toeval dat er vlakbij mijn appartement een schuttersvereniging was; ik ben daar lid van geworden.”
“Regelmatig werd er, bij wat ‘fun shooting’ werd genoemd, op pingpongballen geschoten die op perslucht dansten. Het vervelende was dat steeds

wanneer de pingpongballen geraakt waren, alle wapens moesten worden ontladen (uiteraard) en iemand nieuwe pingpongballetjes in de luchtstroom moest plaatsen. Daar werd het idee geboren om met een druk op de knop pingpongballen via een buis aan te voeren, maar niemand had een idee hoe dat dan mechanisch moest werken want het is natuurlijk wel de bedoeling dat de balletjes er één voor één uit komen. Ik heb toen het een en ander op een bierviltje gekrabbeld, maar uiteindelijk heb ik er niets meer mee gedaan.”

Inmiddels had Ilse zich gevestigd als zelfstandig elektronica-ontwerper, gespecialiseerd in vintage elektronica-onderdelen zoals buizen, nixies en VFD's. Dat had een heel praktische reden – er bestond al heel veel, en ook toen was de concurrentie uit China al groot. De keuze voor een niche-markt ligt dan voor de hand.*

“Dankzij Corona was ik me behoorlijk gaan vervelen; ook mijn schuttersvereniging deed de deur groten-deels op slot. Om toch thuis wat te kunnen oefenen besloot ik om een paar luchtdruk- en airsoftwapens aan te schaffen. Het was ook tijdens de lockdowns een welkom tijdverdrijf.”

* Tijdens de Corona-crisis hebben de Chinezen nagenoeg alle nog bestaande voorraden van Russische buizen opgekocht. Dat is één van de redenen dat de zescijferige nixie-klok van Elektor niet meer leverbaar is.



Figuur 2.
Het pingpongballen-
magazijn.

Bij gebrek aan Nixie-buizen en plexiglas (ook ten gevolge van Corona) wilde Ilse het qua elektronica over een andere boeg gooien, maar het moest wel iets unieks zijn. En daar kwam het idee met de pingpongballen weer om de hoek kijken, met de gedachte om er thuis met luchtdrukwapens op te schieten. Op kermissen is iets dergelijks te zien: schietkramen waar je op pingpongballen kan schieten die op waterstralen dansen (**figuur 1**).

“Momenteel heb ik alleen een werkend prototype van het mechanisch gedeelte opgebouwd. Voor de luchtstroom heb ik een 9BMC24P2Go01-blower van Sanyo Denki gebruikt. Die is helaas niet goedkoop maar levert een stevige luchtstroom en kan via een PWM-sigitaal eenvoudig in toerental geregeld worden. Dat is handig want door de luchtstroom te moduleren kunnen we het pingpongballetje onregelmatig op en neer laten bewegen om het tot een uitdagender doelwit te maken. Door middel van een demonteerbare kast die ook als buffervat dient en een poedertrechter van de chemiegroothandel wordt een krachtige constante luchtstroom gegenereerd. Eventuele vervuiling (stukjes van projectielen) kan zo niet in de blazer terecht komen en je kan de onderkant van de kast openen om hem te reinigen.”

De pingpongballetjes zitten in een schuin opgesteld magazijn gemaakt van een plexiglas buis (diameter 50 mm, binnendiameter 44 mm, **figuur 2**). Door de zwaar-

tekracht rollen ze in de richting van de luchtstroom; er zijn geen beperkingen qua grootte van het magazijn. In het prototype kan dit 11 pingpongballen bevatten maar eventueel is naast een langere buis ook aanvoer via een trechter in het magazijn mogelijk. Een roterende plastic nok zorgt dat de pingpongballen één voor één uit het magazijn rollen. Bij elke heen-en-weer beweging van de nok wordt één pingpongbal vrijgegeven terwijl de overige ballen automatisch geblokkeerd worden. De nok wordt in een basispositie gehouden door een trekveer waarbij de nok door een modelbouwservo wordt bewogen met behulp van een touwtje (zie **figuur 3**).

Figuur 3. Dit
mechanisme zorgt
ervoor dat de ballen
één voor één in de
luchtstroom komen.

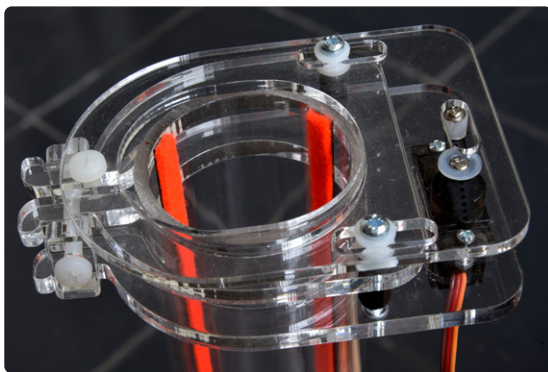


“Een dergelijk mechanisme zou ook de basis kunnen zijn van een Nespresso capsule-automaat. Iets met meerdere buizen naast elkaar. Je geeft dan aan welke capsules je wilt en die vallen dan onderaan in een bakje waar je ze kan uitnemen. Ik speel ook al langer met dat idee, maar daar heb ik nog niks concreets mee gedaan.”

Het balletje dat uit het magazijn rolt [1] komt dan terecht in de luchtstroom boven de opening van de poedertrechter en wordt dan in de verticale plexiglas buis gekatapulteerd. In de buis zijn twee stroken vilt geplakt om te voorkomen dat het balletje ‘ratelend’ in de buis blijft hangen. Dat kan gebeuren als het het balletje in de buis heel snel heen en weer blijft ketsen terwijl de luchtstroom tussen het balletje en de wand van de buis kan ontsnappen.

“Als de verticale buis bovenaan open zou zijn, dan zouden de pingpongballen omhoog gelanceerd worden zonder in de luchtstroom te blijven hangen. Misschien is dat een idee voor een tafeltennis-pingpongballen-dispenser? Om het lanceren van de pingpongballen te voorkomen zit bovenaan de buis een schuif die eveneens door een modelbouwservo wordt bediend (figuur 4). De pingpongballen schieten nu tot boven in de buis tot tegen de schuif. Wanneer de schuif dan langzaam opent, gaat de bal een stukje omhoog om dan even op en neer te dansen tot hij stabiel op de luchtstroom blijft zweven [2]. Vanaf dat moment kan de blazer met een PWM-sigitaal gemoduleerd worden om het gedrag van het balletje wat onvoorspelbaarder te maken. In de praktijk zweeft het balletje zo’n 15 tot 20 cm boven het uiteinde van de schuif. Ik heb ook experimenten zonder verticale buis uitgevoerd, maar dan was het balletje minder stabiel en viel het regelmatig vanzelf uit de luchtstroom.”

►
Figuur 4. Deze schuif voorkomt dat de balletjes gelanceerd worden.



In principe kun je elk denkbaar schiettuig gebruiken om op het pingpongballetje te mikken – tot en met pijl en boog. Het wel zaak om zo goedkoop mogelijke pingpongballen te gebruiken omdat ze geen extreem lang leven zullen hebben; liefst moeten ze ook een contrasterende kleur hebben.

“Meteen na de opbouw van het mechanische gedeelte zijn de problemen met onderdelenschaarste begonnen. De blower is momenteel niet meer verkrijgbaar (althans niet op korte termijn) dus daar zou al een alternatief voor gezocht moeten worden. De rest van de elektronica is niet zo moeilijk omdat in principe alleen twee modelbouwservo's gestuurd moeten worden en een PWM-sigitaal moet worden gegenereerd voor de blower. Op dit moment ligt het project stil, mede door de oorlog in Oekraïne. Ik vind het minder gepast om met al die ellende op TV en in de media een schietspel uit te brengen.”

“Momenteel weet ik niet goed hoe het verder moet en of dit project potentieel heeft. Minimaal zou afstandsbediening nodig zijn. Bij elke druk op de knop zou je dan een nieuwe pingpongbal krijgen, eventueel met een indicatie van de resterende voorraad balletjes. Voor dat laatste zou je IR-reflectiesensoren kunnen aanbrengen op de buis van het magazijn. Andere bijkomende opties zouden een keuze tussen het laten zweven of het ‘lanceren’ van het balletje [2] kunnen zijn of een keuzemogelijkheid om de blower al dan niet te moduleren.”

“Wat ik graag zou willen weten is of hier interesse zou voor zijn en of er eventuele suggesties en wensen zijn voor bijkomende (of andere) functionaliteit. In principe zou het mogelijk moeten zijn om een kit met mechanische onderdelen aan te bieden aangezien alles makkelijk lasergesneden en gelijmd kan worden. Alleen de elektronica is en blijft een pijnpunt tot zeker midden 2023.” ◀

210591-03

Vragen of opmerkingen?

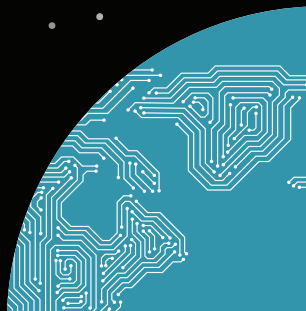
Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de redactie van Elektor via redactie@elektor.com.

WEBLINKS

[1] Video 1 – pingpongballen-magazijn: <https://youtu.be/gTmytGPtHfQ>

[2] Video 2 – zwevend pingpongballetje: https://youtu.be/iH6_5-zcddw

EiC-???



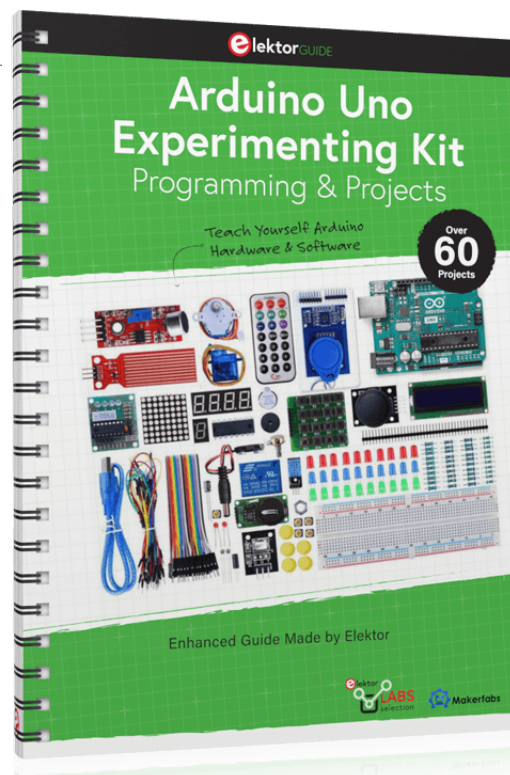
RFID-tag's lezen, en een RFID-deurslot

voorbeeldprojecten uit de Elektor Arduino Experimenteerbundel

Dogan Ibrahim (Verenigd Koninkrijk)

RFID staat voor Radio-Frequency Identification – een draadloze technologie voor applicaties zoals beveiliging, toegangscontrole en het traceren van goederen. In wezen bestaat een RFID-systeem uit een RFID-lezer en een of meerdere tags. Beide hardware-onderdelen zitten in een veelzijdige onderdelenkit, gericht op Arduino-experimenten en ondersteund door een speciaal geschreven handleiding. In dit artikel, waarin Dogan Ibrahim, auteur van vele Elektor-handboeken, een voorbeeld geeft van embedded projectontwikkeling van de grond af, verkent hij de RFID-technologie en krijgt hij hulp van 'ons aller vriend' – de Arduino UNO.

Opmerking van de redactie: Dit artikel is een gedeelte van het 238 pagina's tellende Elektor-handboek "Arduino Uno Experimenting Kit – Programming & Projects". Dit boek is deel van de Elektor Arduino Experimenteerbundel. Het is opgemaakt en enigszins aangepast aan de conventies en pagina-indeling van Elektor Magazine. Omdat het een deel is uit een grotere publicatie, kan dit artikel verwijzen naar passages elders in het boek. Auteur en redactie hebben hun best gedaan om dat te voorkomen en staan voor je klaar als je vragen hebt. De contactgegevens staan in het kader **Vragen of Opmerkingen**.



RFID-technologie maakt gebruik van elektromagnetische velden om gegevens over korte afstanden over te dragen. RFID-systemen worden vooral gebruikt in beveiligingstoepassingen. Ze kunnen bijvoorbeeld worden gebruikt om te zorgen dat een deur uitsluitend door een persoon met de juiste RFID-tag geopend kan worden.

De RFID-lezer die bij de hierboven genoemde Elektor-bundel wordt geleverd (die bundel bestaat uit een handboek en een set onderdelen) is de 'RC522 Module' (**figuur 1**). Hij heeft de volgende specificaties:

- werkfrequentie: 13,56 MHz
- voedingsspanning: +3,3 V
- compatibel met zowel SPI-bus als I²C-bus



Figuur 1. RFID-lezer en RFID-tag.



De RFID-lezer wordt geleverd met pinheaders die in de aansluitingen aan de rand van de print moeten worden gesoldeerd voordat hij kan worden gebruikt.

Project 1: de tag-ID vinden

In dit project leer je hoe je de Tag ID van de specifieke tag in de kit kunt weergeven in de Arduino seriële monitor.

Blokschema: **figuur 2** toont het blokschema van het project, dat eerder illustratief is dan geabstraheerd of functioneel.

Schema: de verbindingen tussen de poorten van het UNO-ontwikkelboard en de RFID-leesmodule zijn als volgt:

Pin RFID Lezer	Poort ontwikkelboard
SDA	10
SCK	13
MOSI	11
MISO	12
IRQ	niet gebruikt
GND	GND
RST	9
3,3 V	3,3 V

Figuur 3 toont het schema van het project. **Let erop dat je de pin van de voedingsspanning niet aan +5 V legt!**

Programmalisting: voordat je de RFID-lezer gebruikt, moet je de RFID-bibliotheek aan je IDE toevoegen. De naam van de bibliotheek is MFRC522, en je voegt hem als volgt toe:

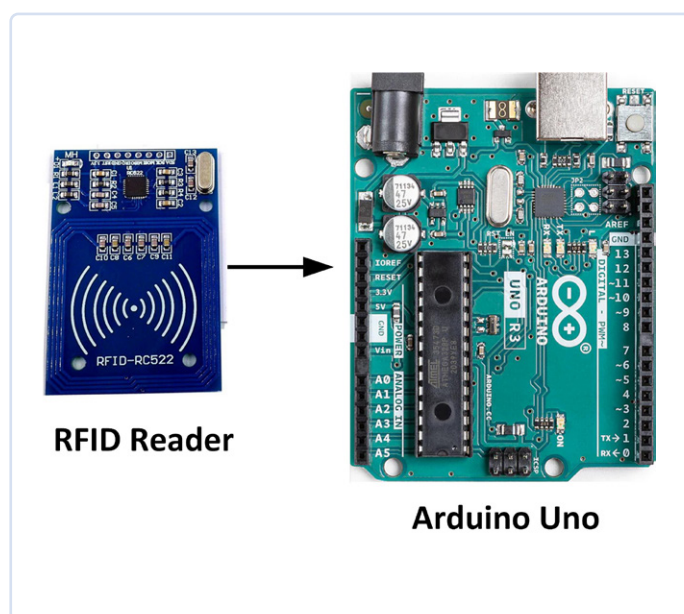
- Download het bestand *rfid-master.zip* naar een map. Je kunt het vinden in het archiefbestand bij het boek dat op de Elektor Store-website staat [1].
- Start de IDE.
- Klik op *Sketch Include* → *Library* → *Add .ZIP Library*.
- Ga naar het opgeslagen .zip-bestand en klik op *Open*.
- Je kunt de bibliotheek nu gebruiken.

De RFID-lezer bibliotheek biedt veel functies die kunnen worden bekeken door het bibliotheekbestand uit te pakken. De belangrijkste functies zijn:

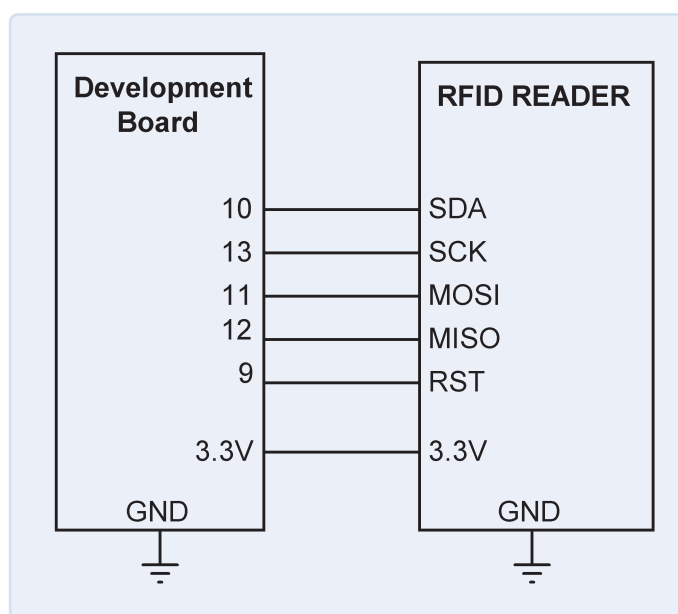
<code>mfrc522.PCD_Init()</code>	De RFID-lezer initialiseren
<code>mfrc522.PICC_IsNewCardPresent()</code>	Zoek naar een RFID-lezermodule
<code>mfrc522.PICC_ReadCardSerial()</code>	Selecteer de te gebruiken RFID-lezer
<code>mfrc522.uid.uidByte[]</code>	Retourneer de Tag-ID in een array
<code>mfrc522.PICC_HaltA()</code>	Stop met lezen (stop PICC)

Het programma *DumpInfo* uit de Arduino IDE kan worden gebruikt om de Tag ID van je lezer te achterhalen. Dat gaat als volgt:

- Start de IDE.
- Klik op *File* → *Examples* → *MFRC522* → *DumpInfo*.
- Compileer het programma en laad het in het ontwikkelboard.
- Start de seriële monitor.
- Plaats de witte tag bovenop de lezer en houd hem daar tot de gegevensweergave in de seriële monitor stopt.



Figuur 2. Blokschema van het tag-lezer project.



Figuur 3. Schema van het tag-lezer project.

```

Firmware Version: 0x92 = v2.0
Scan PICC to see UID, SAK, type, and data blocks...
Card UID: 23 F0 58 A7
Card SAK: 08
PICC type: MIFARE 1KB
Sector Block 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 AccessBits
15 63 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
62 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
14 59 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
58 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
56 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
13 55 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
54 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
53 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
52 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
12 51 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
50 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
49 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
48 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]

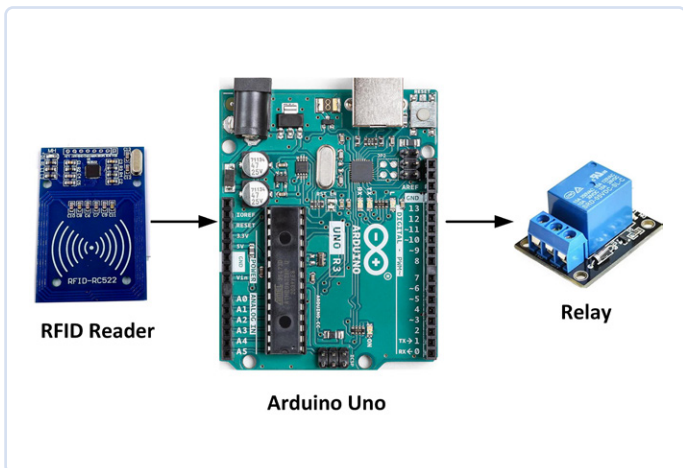
```

Figuur 4: Dump van het tag-geheugen.

Nu zou je iets moeten zien als in **figuur 4**. Dit is het 1KB-geheugen van de kaart inclusief de Tag ID. Het 1KB-geheugen is georganiseerd in 16 sectoren (0..15), waarbij elke sector verder verdeeld is in vier blokken (0, 1, 2, 3). Elk blok kan 16 bytes aan gegevens opslaan (0..15). Dat betekent:

16 sectoren × 4 blokken × 16 bytes =
1.024 bytes data op de kaart (dus 1 KB)

Het derde blok van elke sector (het bovenste blok) wordt de *Sector Trailer* genoemd; dit bevat de *Access Bits*. Deze regelen de lees/schrijftoegang tot de overige blokken in de sector. Daarom zijn alleen de onderste drie blokken (blok 0, 1 en 2) van elke sector beschikbaar voor opslag van gebruikersgegevens. Dit betekent dat je 48 bytes (3 × 16 bytes) per sector van 64 bytes kunt gebruiken. Blok 0 van sector 0 staat bekend als het Manufacturer Block/Manufacturer Data, en bevat de fabrikantgegevens en het ID van de tag. De Tag ID wordt ook weergegeven onder het kopje Card UID in Figuur 4. In dit project is de Tag ID: 23 F0 58 A7.



Figuur 5. Blokschema RFID-toegangscontrole.

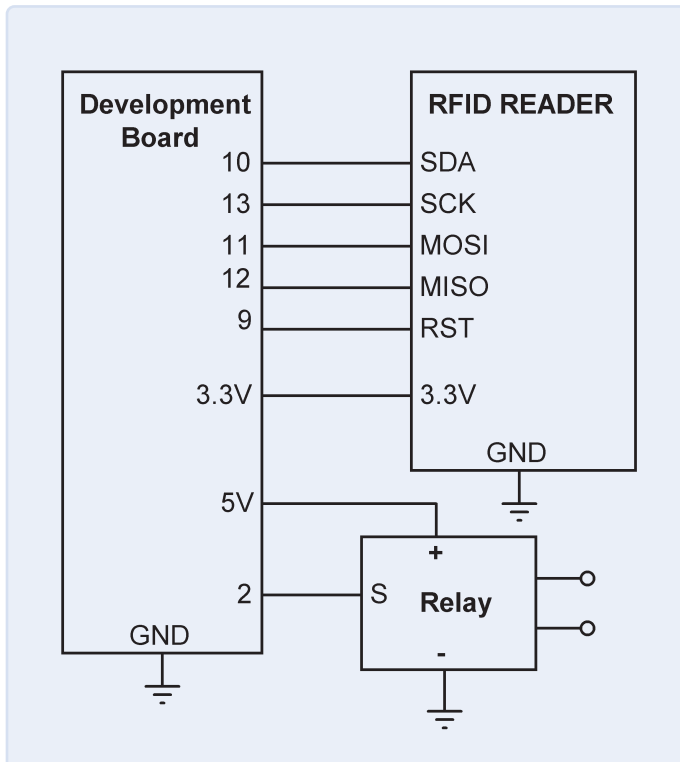
Project 2: RFID-deuropener met relais

In dit project worden zowel de RFID-lezer als het in de kit ingegrepen relais aangesloten op het ontwikkelboard. We nemen aan dat een beveiligde deuropener wordt bediend met een relais dat met een RFID-systeem is beveiligd. De deur kan worden geopend door een geautoriseerde tag-kaart bij de RFID-lezer te houden. Het relais wordt gedurende 15 seconden geactiveerd en daarna gedeactiveerd, zodat de deur weer op slot is.

Blokschema: **figuur 5** toont het blokschema van het project. Ook hier zijn foto's van de echte boards gebruikt in plaats van een geabstraheerde tekening met functieblokken.

Schema: het schema is getekend in **figuur 6**. In tegenstelling tot andere 'schakelende' projecten in het boek (hier niet besproken) wordt een relais aangesloten op poort 2.

Programmalisting: **listing 1** toont het *RFIDLock* genoemde programma dat is ontwikkeld voor het RFID-gestuurde deurslot. Aan het begin van het programma worden de *SPI*- en *MFRC522*-bibliotheken opgenomen. De ValidCard-ID wordt opgeslagen in de *ValidCard*-string, en *RELAY* wordt toegewezen aan Poort 2. In de *setup()*-functie wordt *RELAY* geconfigureerd als uitgang en inactief gemaakt. De *SPI*-bus en de *MFRC522* worden ook geïnitieerd.



Figuur 6. Schema van het toegangscontrole-project.



Listing 1. RFIDLock-programma dat voor dit project is ontwikkeld.

```
//-----  
//                               RFID LOCK SYSTEM  
//                               =====  
//  
// In this program, the RFID card reader is used with a relay. The relay  
// is only activated if an authorized tag is placed near the reader. The  
// relay stays ON for 15 seconds and then turns OFF. The Tag ID of the  
// authorized valid card in this example is: 23 F0 58 A7  
//  
// Author: Dogan Ibrahim  
// File  : RFIDLock  
// Date  : May, 2022  
//-----  
#include <SPI.h>  
#include <MFRC522.h>  
  
#define SS_PIN 10  
#define RST_PIN 9  
MFRC522 mfrc522(SS_PIN, RST_PIN);    // Create MFRC522 instance  
String ValidCard = "23F058A7";        // Valid Tag ID  
String TagID = "";  
int RELAY = 2;                        // RELAY at port 2  
byte i;  
  
void setup()  
{  
    pinMode(RELAY, OUTPUT);            // RELAY is output  
    digitalWrite(RELAY, LOW);          // Deactivate RELAY  
    SPI.begin();                      // Initiate SPI bus  
    mfrc522.PCD_Init();                // Initiate MFRC522  
}  
void loop()  
{  
    if (!mfrc522.PICC_IsNewCardPresent()) // Look for card  
    {  
        return;  
    }  
  
    if (!mfrc522.PICC_ReadCardSerial())    // Select the card  
    {  
        return;  
    }  
    TagID = "";  
  
    for (i = 0; i < 4; i++)                // Read 4 byte Tag ID  
    {  
        TagID.concat(String(mfrc522.uid.uidByte[i], HEX));  
    }  
  
    TagID.toUpperCase();                  // Convert to upper case  
    mfrc522.PICC_HaltA();                 // Stop reading  
    if(TagID == ValidCard)                // Valid card?  
    {  
        digitalWrite(RELAY, HIGH);        // RELAY ON  
        delay(15000);                     // Wait 15 seconds  
        digitalWrite(RELAY, LOW);         // RELAY OFF  
    }  
    else  
        digitalWrite(RELAY, LOW);         // RELAY OFF  
}
```



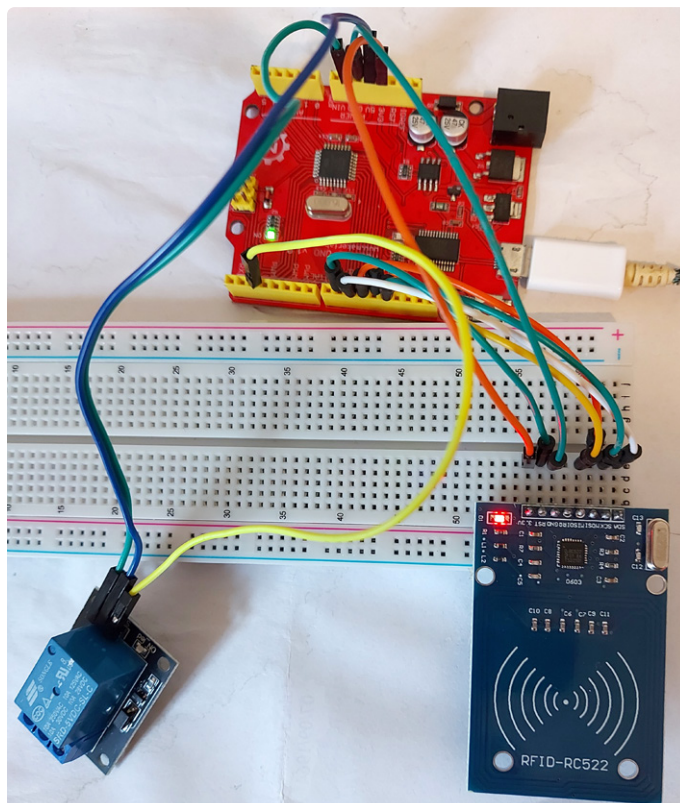

De rest van het programma draait in de hoofd-programmalus. Hier wacht het programma tot een kaart in de buurt van de lezer wordt gehouden, waarna de kaart wordt herkend en een `for`-lus wordt gestart om de 4-byte Tag ID in de variabele `TagID` in te lezen. Daarna wordt `TagID` vergeleken met de geautoriseerde Tag ID in `ValidCard`. Als die hetzelfde zijn, is de gebruiker geautoriseerd en wordt het relais gedurende 15 seconden geactiveerd. Als de kaart niet geldig is, blijft het relais gedeactiveerd. Dit hele proces wordt continu herhaald.

Figuur 7 toont tenslotte het project zoals dat op breadboard is gerealiseerd. ◀

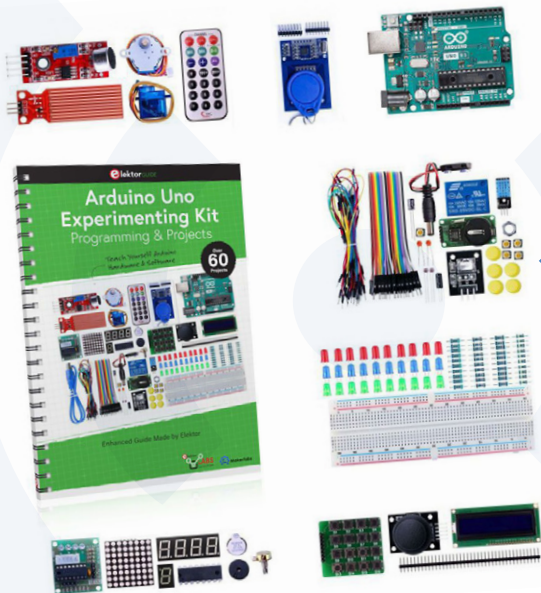
vertaling: Hans Adams — 230066-03

Vragen of opmerkingen?

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



Figuur 7. Het toegangscontrole-project op breadboard.



Gerelateerde producten

- > **Arduino Uno Experimenting Bundle (SKU 20340)** bestaande uit:
 - Elektor Guide: Arduino Uno Experimenting Kit - Programming & Projects (238 pagina's)
 - Makerfabs onderdelenkit inclusief een echte Arduino Uno R3.

www.elektor.nl/arduino-uno-experimenting-bundle



WEBLINK

[1] Software-archief bij de experimenteerbundel: <http://www.elektor.nl/arduino-uno-experimenting-bundle>

HF-stroomprobe voor oscilloscoop

HF-stroommetingen gemakkelijk gemaakt

Roberto Visentin (Italië)

Moet je de antennestroom van je HF-zender meten? Of, zoals ik, de primaire stroom van je Tesla-spoel? Deze projecten vereisen enig denkwerk vooraf. Ik geef hier enkele ontwerprichtlijnen en een voorbeeld uit de praktijk.



Figuur 1. Een close-up van mijn HF-stroomprobe.

In veel gevallen zijn stroommetingen zonder DC-component nuttig. Het betreft dan meestal CT's (current transformers, stroomtransformatoren) voor wisselstroom. Dit artikel gaat over het ontwerp van stroomtransformato-

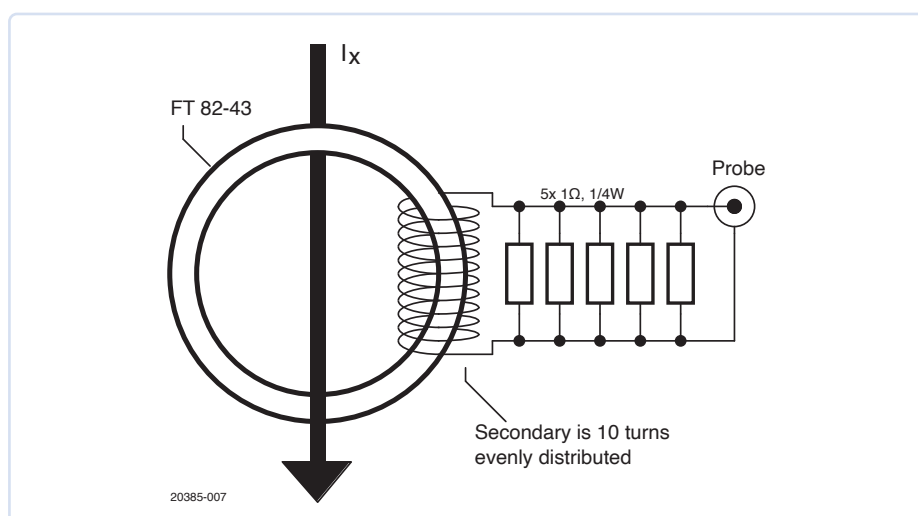
ren voor gemiddelde tot hoge frequenties, die heel eenvoudig te bouwen zijn. De gepresenteerde formules zijn ook geldig voor AC-nettransformatoren.

Het principe van de probe

De probe van **figuur 1** is ontworpen om tot 50 A piek te meten in een frequentiebereik van 7 kHz tot tientallen MHz. Het schema in **figuur 2** is vrij eenvoudig: de draad waarvoor de stroom moet worden gemeten, wordt door de ringkern gevoerd; dat is een gewone Amidon-kern type FT 82-43 die goed presteert tot ten minste 50 MHz.

De secundaire winding bestaat uit tien gelijkmatig over de kern verdeelde windingen. Gebruik indien mogelijk een niet te dunne draad, maar dat is niet verplicht. Door de wikkelverhouding van 1:10 is de maximale stroom in de secundaire 5 A piek.

De secundaire winding is belast met 0,2 Ω in de vorm van een parallelschakeling van vijf weerstanden van 1 Ω . Bij een piekstroom van 5 A bedraagt de piekspanning over deze weerstanden 1 V, prima geschikt voor



Figuur 2. Het simpele schema van mijn HF-stroomprobe.

metingen met een oscilloscoop. Voor een sinusvormige stroom is de gemiddelde dissipatie in de weerstanden $R \cdot I^2 = R \cdot I_p^2 / 2 = 2,5 \text{ W}$ of $0,5 \text{ W}$ per weerstand. Een continue sinusvormige stroom van 50 A_p kan alleen worden gemeten met weerstanden van $0,5 \text{ W}$ of groter. Maar bij gepulste golfvormen of als er zeer korte metingen worden verricht, zijn weerstanden van $0,25 \text{ W}$ voldoende. Dat was mijn keuze omdat ik het ontwerp compact wilde houden voor betere HF-prestaties. Eerlijkheidshalve moet ik ook toegeven dat ik deze weerstanden toevallig bij de hand had...

Gebruik

Figuur 3 toont een typische toepassing met een oscilloscoop-probe en een BNC-adapter voor probes. De schakeling kan ook worden gebruikt met een directe coaxverbinding met de 'scoopgang, omdat 1 V_p ideaal is voor gebruik met een oscilloscoop in 1x-modus: in dit geval wordt het gebruik van een korte kabel aanbevolen om reflecties in de betreffende frequentieband te voorkomen, omdat de coax aan beide zijden niet correct is afgesloten. Nog beter is het om de coaxkabel aan 'scoopzijde af te sluiten met zijn karakteristieke impedantie. Veel moderne oscilloscopen

bieden de mogelijkheid om de ingangsimpedantie op 50Ω in te stellen, zodat dit bijzonder gemakkelijk is. In dit specifieke geval moet men bedenken dat de meting enigszins verkeerd geschaald zal zijn, vanwege de parallelschakeling van de 50Ω belasting en de $0,2 \Omega$ van de probe (de totale weerstand wordt $0,1992 \Omega$, wat een schaalfactor geeft van $50,2 \text{ A/V}$). Wat moet worden vermeden, is de probe van de 'scoop rechtstreeks met de klemmen op de weerstanden aan te sluiten zonder gebruik te maken van de BNC-connector, want bij het meten van hoge HF-stromen zal zelfs de minimale niet-afgeschermde lus in de probes artefacten toevoegen aan de metingen.

Berekeningen

Het ontwerp van de stroomtransformator is niet ingewikkeld, maar enkele elektromagnetische formules zijn onontkoombaar. Eerst de belastingsweerstand R_L , die zo klein als praktisch mogelijk moet zijn om het erdoor geïntroduceerde vermogensverlies te minimaliseren. De te meten kring 'ziet' minstens $R \cdot n^2$, waarbij n de wikkilverhouding is (1:10) en R de som van R_L ($0,2 \Omega$) en van de secundaire draadweerstand (enkele mΩ). Zoals al opgemerkt, is het van groot belang dat de

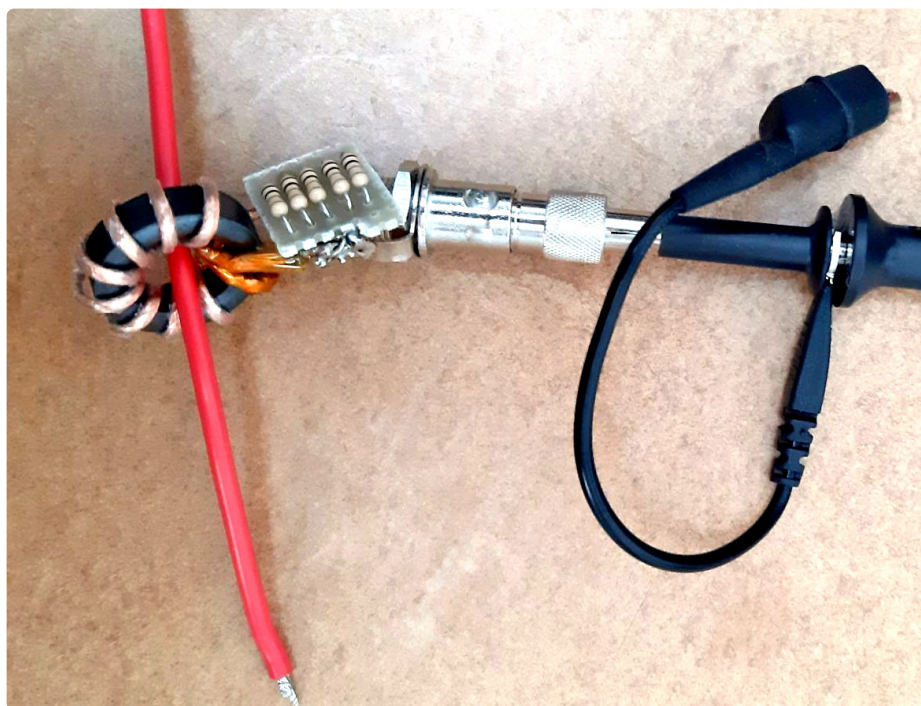
secundaire zijde gelijkmatig wordt gewikkeld, omdat de te testen kring anders enige strooi-inductie in serie vertoont. Aan de andere kant, als we een te kleine waarde voor R_L kiezen, zullen we ook een zeer kleine te meten spanning hebben, wat ruis in het 'scoopbeeld geeft. Tenslotte moet R_L groter zijn dan de weerstand van de secundaire winding. In mijn geval koos ik $0,2 \Omega$ zodat ik 1 V zou krijgen bij 5 A (50 A op de primaire), wat $2 \text{ m}\Omega$ toevoegt aan de te testen kring.

Het aantal secundaire windingen, n , bepaalt de transformatieverhouding. In het geval van een hoogfrequente CT moet deze n klein worden gehouden om eigenresonantie door strooicapaciteiten in combinatie met een hoge zelfinductie te voorkomen. Bij CT's voor het lichtnet is de frequentie vrij laag (50 of 60 Hz) en daarom is $n = 1000$ een gebruikelijke waarde. Machten van 10 zijn gebruikelijk, zodat de stroom-transformatieverhouding eenvoudig blijft, maar andere waarden zijn mogelijk. De hoogste bruikbare frequentie voor een ferriet ringkern-CT hangt af van de:

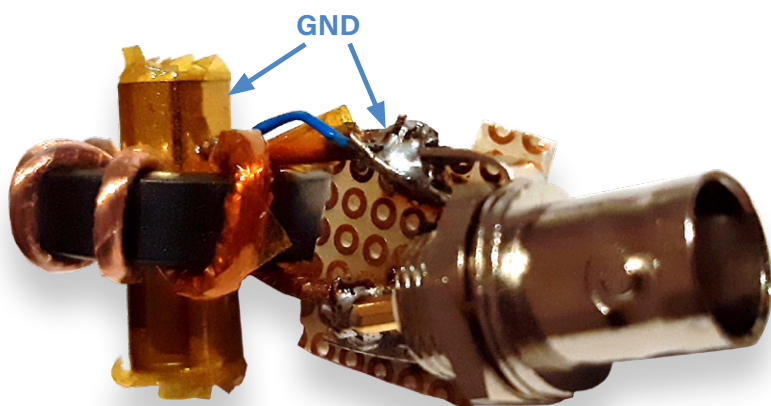
- > prestaties van het ferrietmateriaal;
- > eigenresonantie van de secundaire wikkeling;
- > weerstands-zelfinductie (weerstand zelf en aansluitingen).

Een ontwerp als het mijne kan gemakkelijk tot enkele tientallen MHz functioneren indien een geschikt ferriet, zoals het materiaal '43' van Amidon/Fair-Rite, wordt gebruikt. Kernen met hoge permeabiliteit voor EMI-onderdrukking kunnen ook worden gebruikt, maar dan met een veel lagere maximale frequentie. Kernen met een lage permeabiliteit zoals worden gebruikt voor vermogenssmoorspoelen en hoge-Q-spoelen zijn niet aan te bevelen, omdat hun zelfinductie per winding te laag is, hetgeen van invloed is op het volgende punt. De keuze van de ferrietkern is ook van invloed op de laagste bruikbare frequentie, en wel om twee redenen:

- > voor een goede meting moet de reactantie van de secundaire veel groter zijn dan R , omdat deze ook als belasting werkt. Aangezien voor de reactantie geldt $X = 2\pi \cdot f \cdot L$, heeft dit gevolgen voor de lagere frequenties. In de praktijk willen we aan de lage kant $X > 10 \cdot R_L$;



Figuur 3. De HF-stroomsonde in de praktijk.



Figuur 4. Verbeterde probe met elektrostatiesche afscherming.

- we moeten verzadiging van de kern vermijden, die optreedt bij hoge stroom en lage frequentie. Verzadiging van ferrieten treedt op bij een fluxdichtheid $B = 0,25 \dots 0,3$ T, maar om niet-lineaire verschijnselen te voorkomen, moeten we onder 0,2 T blijven bij de maximale stroom en de minimale frequentie.

Om beide bovenstaande redenen zijn voor lage bruikbare frequenties met een klein aantal windingen en een hoge stroom grote kernen nodig. De belangrijkste kernparameters zijn A_L , de inductantie per winding, meestal uitgedrukt in nH/n^2 voor HF-kernen, en de kerndoorsnede S , die gemakkelijk kan worden berekend uit de afmetingen van de kern. In het geval van de kern Amidon FT 82-43 is $A_L = 470$ nH/n^2 en is de doorsnede $24,6$ mm^2 (gemakkelijk te berekenen uit buitendiameter, binnendiameter en hoogte).

De inductantie van 10 windingen is dus 470 $nH \times 10^2 = 47$ μH (vergeet niet dat de zelfinductie afhangt van n^2). Als we aannemen dat $X = 10 \cdot R \approx 10 \cdot R_L = 2$ Ω , krijgen we (door de formule voor de reactantie om te keren) $f > 6,8$ kHz als lage afsnijfrequentie. Dit kan aanvaardbaar zijn of niet, afhankelijk van de toepassing, en een grotere kern, met een grotere A_L , zal een lagere afsnijfrequentie hebben. In mijn geval, voor het meten van een 'muzikale' Tesla-spoel, was ik geïnteresseerd in frequenties boven 500 kHz , dus deze kern voldeed prima.

Met betrekking tot verzadiging geldt in het sinusvormige regime voor de grootte van het veld B (de magnetische fluxdichtheid) $abs(B) = V_p / (n \cdot 2\pi \cdot f \cdot S)$ is, waarbij V_p

de piekspanning over de wikkeling is. Deze vergelijking kan worden afgeleid door de twee uitdrukkingen voor de magnetische flux $\Phi_c = L \cdot I = n \cdot B \cdot S$ gelijk te stellen en I te vervangen door de absolute waarde van de magnetiserende wisselstroom $abs(I) = V / (2\pi \cdot f \cdot L)$. Indien V en n in eerdere ontwerpfasen zijn vastgelegd, is de enige manier om een lagere f te bereiken het gebruik van een grotere S : een grotere torus.

In mijn geval is $V_p = 1$ V, $n = 10$, $S = 24,6$ mm^2 (let op de juiste eenheden), dus hebben we $B < 0,2$ T voor $f > 3,2$ kHz , wat prima is (althans voor mijn toepassing). Het is misschien verwarrend dat de stroom niet in de berekening voorkomt: hij gaat eigenlijk schuil achter V , omdat we weten dat 1 V overeenkomt met 50 A.

Verbeteringen

Capacitieve koppeling tussen primaire en secundaire windingen kan metingen verstoren bij de hoogste nog bruikbare frequenties, of zelfs bij gemiddelde frequenties als de primaire geleider een hoge RF-spanning voert. Het ontwerp kan worden verbeterd door een elektrostatiesche afscherming toe te voegen die

deze capacitieve koppeling voorkomt: in de praktijk wordt de primaire draad daartoe door een klein stukje metalen buis (typisch koper of messing) gevoerd dat verbonden is met de uitgangs-GND van de secundaire wikkeling, zoals te zien in **figuur 4**. Dit verandert niets aan de magnetische koppeling, maar blokkeert het elektrische veld.

Conclusie

Dit voorbeeld van een HF-stroomtransformator met de belangrijkste ontwerpcriteria bewijst dat deze materie minder complex is dan zij in eerste instantie lijkt. Ik hoop dat de hier gepresenteerde overwegingen en formules nuttig zijn om de omgang met ringkernen te vereenvoudigen, en als basis dienen voor je eigen ontwikkelingen. ◀

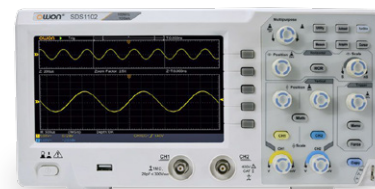
200385-03

Over de auteur

Roberto Visentin is een onlangs gepensioneerd elektrotechnicus die aan elektronica en besturingssystemen voor maritieme toepassingen en onderwaterrobotica heeft gewerkt. Hij is nog steeds actief als freelance consultant, maar heeft nu gelukkig ook meer tijd om hobbyprojecten te ontwikkelen in zijn elektronica-lab thuis.

Vragen of opmerkingen?

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



Gerelateerde producten

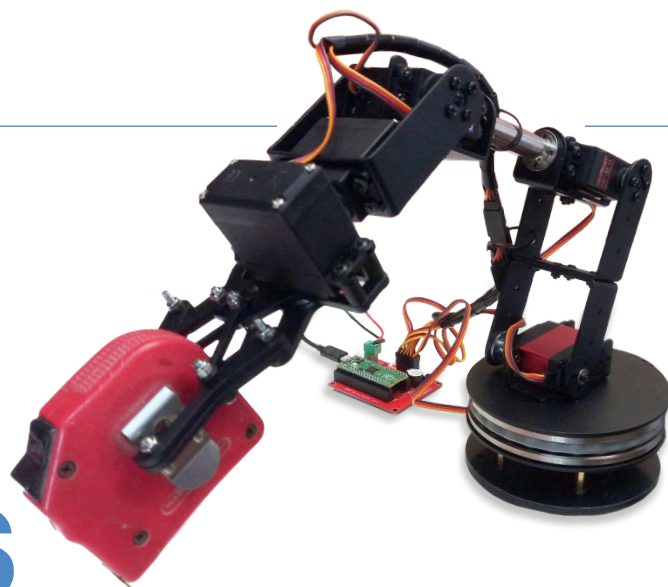
- **OWON SDS1102 2-channel oscilloscope (100 MHz) (SKU 18782)**
www.elektor.com/18782
- **OWON XSA810 Spectrum Analyzer (1 GHz) (SKU 19714)**
www.elektor.com/19714



Niet voor watjes –

Makerfabs Robot Arm Kit

met de Raspberry Pi Pico en MicroPython



Clemens Valens (Elektor)

Een robotarm die in veel richtingen kan bewegen en objecten kan oppakken dankzij zes gewrichten is leuk om mee te spelen, zeker wanneer hij eenvoudig kan worden aangestuurd met MicroPython op een Raspberry Pi Pico. Met deze kit kun je zelf een krachtige volledig metalen robotarm bouwen.

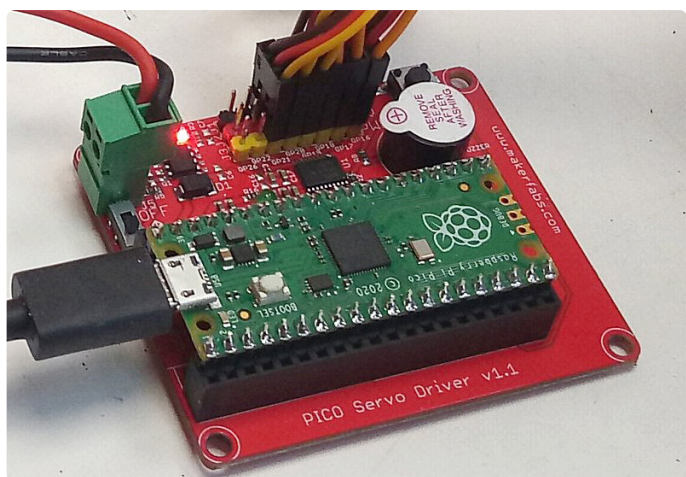
Wat is het?

De 6 DOF Robot Arm met de Raspberry Pi Pico[1] is een robotarm bestuurd door een Raspberry Pi Pico microcontrollerboard. DOF is de afkorting van Degree(s) of Freedom (vrijheidsgraden), wat in het geval van een robotarm betrekking heeft op het aantal gewrichten. De arm die hier wordt beschreven heeft er zes, waardoor deze gelijktijdig zes verschillende onafhankelijke bewegingen kan maken. Elke DOF betekent een aparten servo.

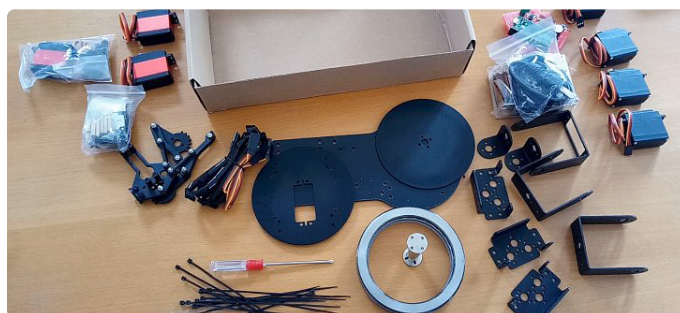
De arm wordt bestuurd door een Raspberry Pi Pico gemonteerd op een klein uitbreidingsprintje (**figuur 1**) dat zorgt voor de verbindingen met de servo's, de voeding, een zoemer en een drukknop. De Pico, of beter gezegd de RP2040 MCU daarop, is ontworpen om in MicroPython te worden geprogrammeerd, maar dit is natuurlijk niet verplicht. De driver en de voorbeelden zijn echter wel in MicroPython geschreven.

Alles is inbegrepen

De robotarm wordt geleverd als een complete set onderdelen (**figuur 2**). Behalve een computer voor de programmering en een micro-USB kabel heb je niets anders nodig. Een voeding (5 VDC, 6 A) is inbegrepen. Een netsnoer is ook inbegrepen, maar met een



Figuur 1. Een Raspberry Pi Pico bestuurt de zes servo's van de robotarm.



Figuur 2. De 6 DOF robotarm wordt geleverd als compleet bouw pakket.



Amerikaanse stekker. Dus afhankelijk van waar je woont moet je deze mogelijk vervangen of een verloopstekker gebruiken (zoals ik deed). Er zit zelfs een kleine schroevendraaier bij, maar ik gebruikte voor het monteren van de arm liever mijn eigen gereedschap. De kit heeft heel wat onderdelen en nog meer schroeven en moeren, maar niet alles wordt gebruikt. Met uitzondering van de acryl-basisplaat en de stalen 'arm' zijn alle onderdelen gemaakt van aluminium. Indien je alles goed in elkaar zet, krijg je geen wiebelig plastic speeltje, maar een sterke, solide metalen arm. Ondanks dat alles van aluminium is, weegt de geassembleerde arm 1,3 kg (zonder de voeding).

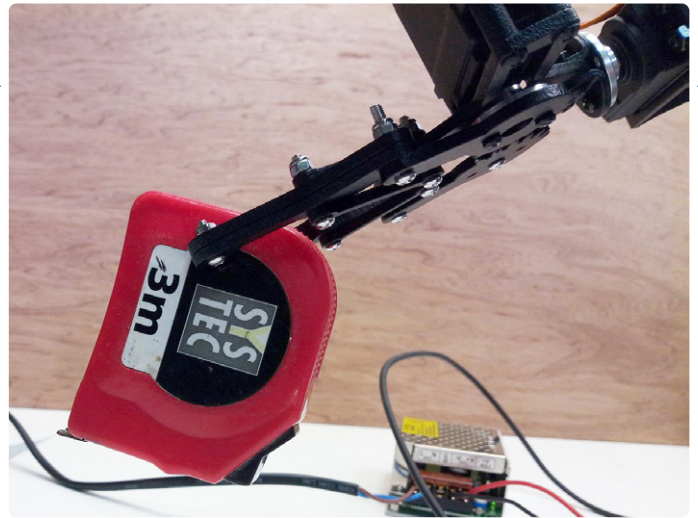
Leerzaam hulpmiddel

De 6 DOF-robotarm is vooral een hulpmiddel om te leren programmeren in MicroPython. Het is een leuk apparaat dat op veel manieren kan bewegen, en hij kan kleine voorwerpen zoals zijn eigen voeding oppakken (ca. $6 \times 4 \times 8,5$ cm). Zoals te zien op een demo-video[2] kun je hem bijvoorbeeld gebruiken om het Torens van Hanoi-spel te spelen. Hij is niet geschikt voor industriële toepassingen of zeer nauwkeurige taken. De arm heeft wel grip, maar ik zou hem met een werktuig toch niet vertrouwen (**figuur 3**).

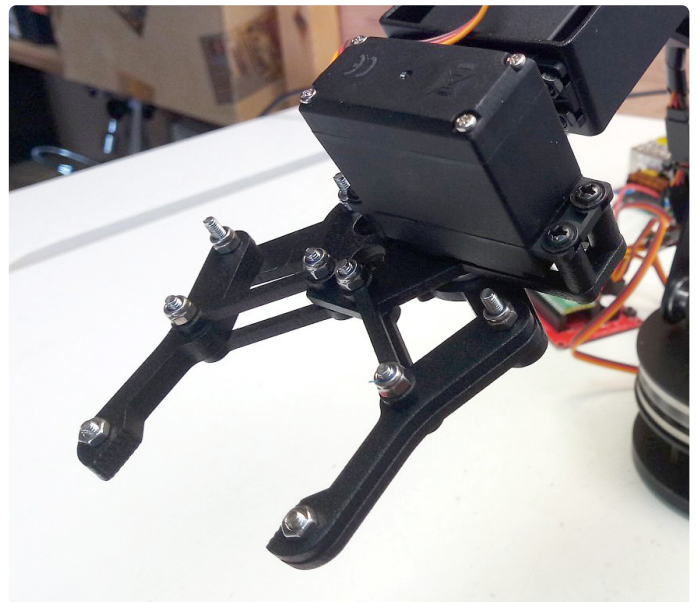
Wat je misschien vooraf wilt weten

Het monteren van de robotarm is niet voor watjes. De handleiding [3] (niet inbegrepen, en je mag zelf uitzoeken waar je hem kunt vinden) is alleen in het Chinees en bovendien onduidelijk en onvolledig. Er is ook een korte montagevideo (animatie) [4], maar deze is ook onvolledig. Toch kan het met een beetje geduld en gezond verstand wel lukken omdat het geen erg ingewikkeld ontwerp is. Het enige complexe onderdeel, de gripper, is al voormonteed (**figuur 4**). Er zijn echter wel een paar zaken die je tevoren moet weten (en waar ik door schade en schande achter ben gekomen):

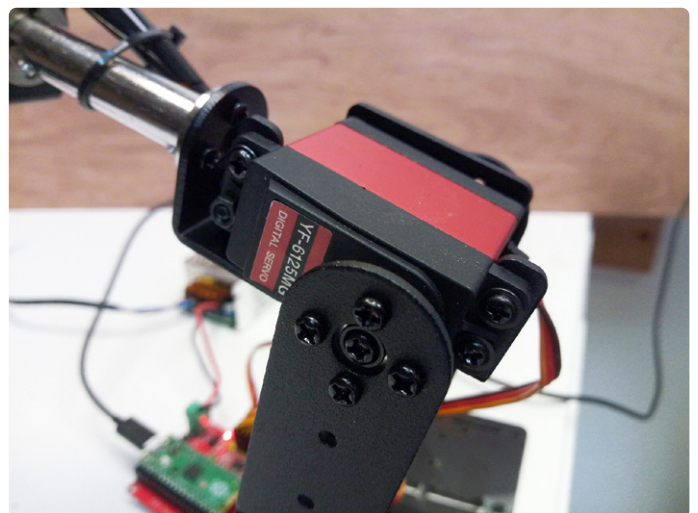
- > Wees erop voorbereid om de boel uit elkaar te halen zodat je het weer op de juiste manier in elkaar kunt zetten. Twee keer bouwen geeft waarschijnlijk de beste resultaten.
- > Begin onderaan, dus bij de bodemplaat met het roterende platform, en ga van daaruit omhoog.
- > Het grote lager van de roterende basis bestaat uit drie delen. De ene van de twee mogelijke configuraties draait iets beter dan de andere, maar uiteindelijk lijkt dat er helemaal niets toe te doen omdat de basis voornamelijk als een dood gewicht functioneert.
- > Gebruik eerst de twee rode YF-6125MG-servo's (**figuur 5**). Mijn kit bevatte er drie (één reserve), maar in de handleiding en video worden deze overal gebruikt. Deze servo's zijn het krachtigst en kunnen het meeste gewicht dragen; daarom moeten ze op het onderste deel van de arm worden gemonteerd.
- > Zorg ervoor dat alle servo's in de middenpositie staan voordat je ze monteert. Bij de zwarte MG 996R-servo's kun je dit met de hand en op het oog doen met een soort servopointer/arm (meegeleverd in de kit). De rode YF-6125MG-servo's hebben echter geen eindstops, zodat je niet kunt zien wanneer ze in de middenpositie staan. Ik heb daarom de Pico gebruikt om ze al doende op nul te zetten (dat wil zeggen tijdens het monteren van de arm). Dit klinkt misschien ingewikkeld, maar is eigenlijk best eenvoudig. Ik zou zelfs willen adviseren



Figuur 3. De arm kan een werktuig vasthouden, maar ik zou hem daarmee niet vertrouwen.



Figuur 4. De gripper is al voormonteed, je hoeft alleen nog de servo te monteren.



Figuur 5. De YF-6125MG-servo's zijn krachtig en dienen het grootste deel van het gewicht van de arm te dragen.

om de montage te beginnen met de installatie van de MicroPython-ontwikkelomgeving en te proberen eerst de demo's uit te voeren met de nog niet gemonteerde servo's. Als je eenmaal door hebt hoe je de servo's via MicroPython aanstuurt, kun je ze op nul zetten en op de arm monteren.

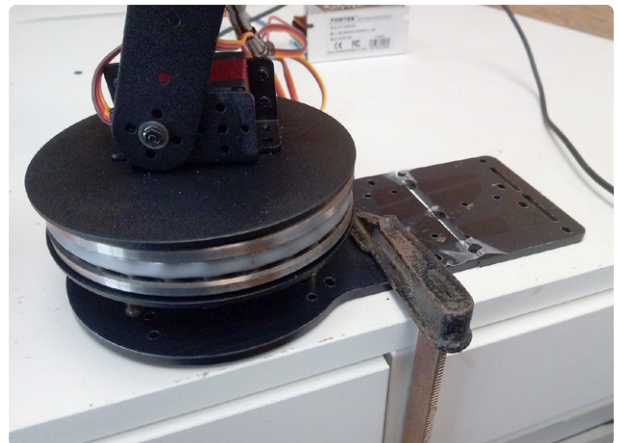
- Bij alle montageschappen wordt de servo als laatste gemonteerd. Deze dienen over het algemeen te worden bevestigd met de schroeven met sluitring (model 'Mexicaanse hoed').
- Zorg ervoor dat je alle schroeven goed vastdraait, want dit is moeilijk nadat de arm is gemonteerd.
- De grijper is voorgemonteerd maar een beetje gammel, zodat ik de zelfborgende moeren wat verder aangedraaid heb. Het gevolg was dat de servo die de grijper bestuurt is doorgebrand, omdat de grijper blijkbaar te stroef was geworden om te bewegen (hoewel ik hem wel handmatig kon verplaatsen). Gelukkig bevat de kit twee reserve-servo's.
- Er is geen bedradingsschema, er zijn zes servo's en de header van de servo heeft acht posities. Gelukkig is deze kleur-gecodeerd dus is het eenvoudig om de polariteit goed te krijgen. Sluit de servo's beginnend bij de onderste aan. De laagste servo (de roterende basis, 'o' of 'A' in de software) wordt aangesloten op GPIO16, de hoogste (de grijper, '5' of 'F' in de software) gaat naar GPIO21. De software maakt het mogelijk om de GPIO-pinnen opnieuw toe te wijzen en ook de draairichting van elke servo kan door de gebruiker worden gedefinieerd, dus je hebt hier mogelijkheden om de boel te corrigeren.
- De bodemplaat heeft veel gaten, maar niet voor het bevestigen van het Pico-printje of de voeding.
- Zorg ervoor dat de voeding 5 V levert voordat je die op de arm aansluit. De uitgangsspanning kan worden ingesteld tussen ongeveer 4,5 V en 6,5 V.

Pas op!

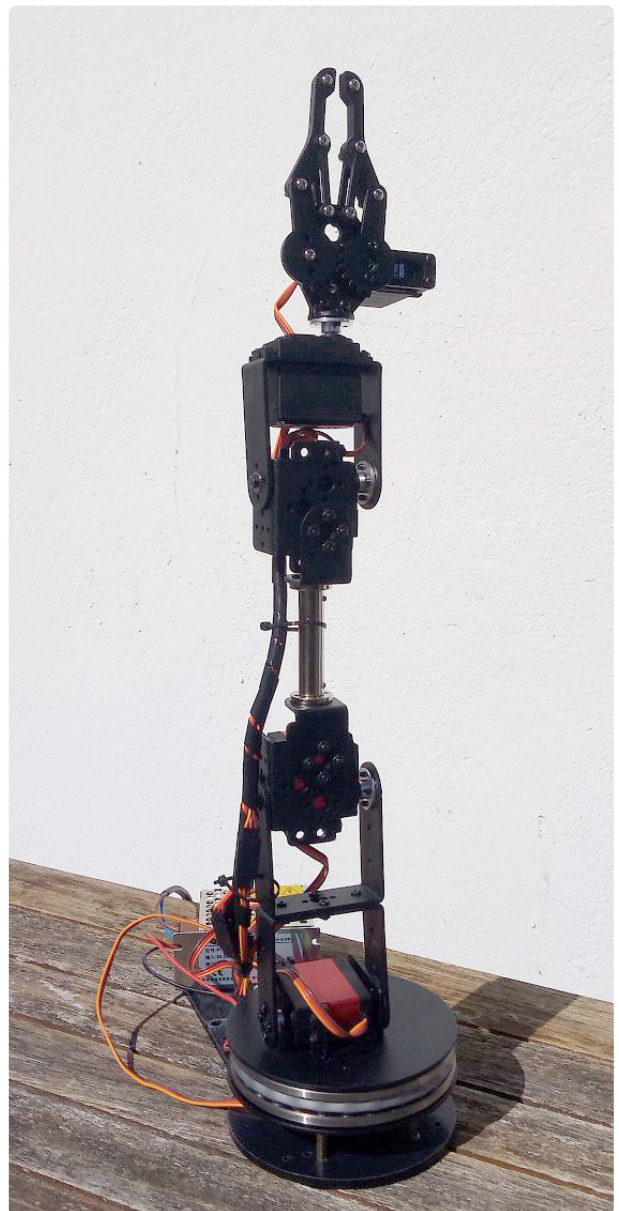
Monteer de robotarm niet en voer geen demo's zomaar uit zonder eerst elke servo te hebben getest. De arm kan ronddraaien en wild bewegen, spullen van je werkbank vegen of je in het gezicht raken. De servo's zijn krachtig! Ik slaagde erin om de arm zijn eigen 4 mm dikke basisplaat te doen knappen omdat een servo omlaag tegen het werkblad drukte in plaats van omhoog te bewegen. Vooral de rode YF-6125MG-servo's zijn behoorlijk snel, dus pas op. Ik raad daarom ten eerste aan om de arm op het werkblad te fixeren (**figuur 6**), of op een zware plaat, en de ruimte eromheen vrij te houden. Vergeet niet dat de rode YF-6125MG-servo's geen eindstops hebben en de verkeerde kant op kunnen bewegen om de beoogde positie te bereiken. Dit kan gebeuren wanneer de arm wordt neergeklapt en vervolgens wordt ingeschakeld. Ik raad daarom ook ten eerste aan om de arm rechtop te zetten voordat je de stroom inschakelt (**figuur 7**).

Niet voor watjes

De 6 DOF Robot Arm met Raspberry Pi Pico van Makerfabs [6] is geen kit voor watjes. Het correct in elkaar zetten vergt geduld en doorzettingsvermogen. Hoewel de arm vrij eenvoudig in het gebruik is, dankzij de driver en demo die in MicroPython zijn geschreven, dient hij stap voor stap te worden getemd om ongelukken te voorkomen. De servo's zijn snel en sterk en de arm is 44 cm lang wanneer



Figuur 6. Je kunt veiliger experimenteren door de robotarm aan het werkblad te fixeren.



Figuur 7. Rechtop is de veilige startpositie.



Deze bespreking op video:
<https://youtu.be/yb54d4vDPq0>



hij volledig is gestrekt (50 cm inclusief de roterende basis). Het vastschroeven aan het tafelblad in het midden van een vierkante meter waar niets staat wordt ten zeerste aanbevolen. Je wilt toch niet dat hij je laptop verbrijzelt of je koffiemok van tafel veegt? 

220232-03

Vragen of opmerkingen?

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



Gerelateerde producten

- **Makerfabs 6 DOF Robot Arm with Raspberry Pi Pico (SKU 20130)**
www.elektor.com/20130
- **Elektor Raspberry Pi Pico Experimenting Kit (SKU 19834)**
www.elektor.com/19834

WEBLINKS

- [1] 6 DOF Robot Arm bij Elektor: <https://www.elektor.nl/makerfabs-6-dof-robot-arm-with-raspberry-pi-pico>
- [2] Demovideo: <https://youtu.be/8dB1W6Jd07g>
- [3] Bouwbeschrijving [zip-bestand]: <https://www.makerfabs.com/desfile/files/Raspberry-Pi-PICO-6-DOF-Robot-Arm.zip>
- [4] Video van de montage: <https://youtu.be/NqL9n3avBbY>
- [5] MicroPython driver en voorbeeld: https://github.com/Makerfabs/PICO_Merchanical_Hand_Driver
- [6] Makerfabs-website: <https://makerfabs.com/raspberry-pi-pico-6-dof-robot-arm.html>

Zij vertrouwen ons, u ook?

Wij houden van elektronica en van projecten, en doen ons uiterste best om aan de behoeften van onze klanten te voldoen.

De Elektor Store:
**'Nooit duur,
 altijd verrassend'**

Bekijk meer beoordelingen op onze Trustpilot-pagina: www.elektor.nl/TP
 of oordeel zelf door een bezoek te brengen aan onze Elektor Store, www.elektor.nl

"Goed en snel – Snel, correct, goeie prijs"

★★★★★ by Roger Vöhr

"Bouwpakketje besteld, volgende dag geleverd – Wo 20 juli een FG085 miniDDS Function Generator Kit besteld. Do 21 juli geleverd. Niet alleen vlot geleverd, ook de communicatie rondom het bestelproces is prima. Vanzelfsprekend vervult ..."

★★★★★ by M. Hout

"jullie doen gewoon goed werk!, duidelijke en goede verzending"

★★★★★ by Benny

"Zeer vlotte afhandeling van de bestelling en zeer concurrentiële prijs."

★★★★★ by Walter Lippers





Generatieve KI

wie heeft dit eigenlijk gemaakt?

Priscilla Haring-Kuipers (Nederland)

Het draait allemaal om KI. DALL-E en ChatGPT – beide van OpenAI – hebben velen in hun ban gebracht en generatieve KI in de samenleving geïntroduceerd.

K staat voor kunst

Het grootste debat rond visueel generatieve KI's (of AI's) zoals DALL-E [1] gaat over de kunst waarop de KI trainde zonder de oorspronkelijke kunstenaars te vragen. Voor mensen vinden we het prima om geïnspireerd te worden door kunst. We vinden het zelfs goed dat beginnende kunstenaars kunst kopiëren als uitgangspunt, maar het algemene idee is dat je dan overgaat tot het maken van origineel werk. Waarom kan een systeem online-kunstwerken niet op deze manier gebruiken? Een van de redenen is dat dit niet aan de orde was toen de kunstenaar zijn kunst online zette. Ze stelden hun kunst niet beschikbaar voor training en niemand vroeg hen of ze hun kunst op deze manier mochten gebruiken. Ze zijn zijn plotseling met andere spelregels geconfronteerd en nu moeten we allemaal samen uitzoeken wat we als nieuwe regels accepteren en hoe ze moeten worden toegepast.

Een andere reden is dat KI niet door kunst wordt geïnspireerd; het construeert een dataset waarmee het een model traint. We begrijpen intelligentie meestal door de bril van onze eigen ervaringen en verwachten dat KI op vrijwel dezelfde manier werkt als menselijke intelligentie. Dat is niet het geval. Wij zijn multi-modale wezens met directe en indirecte verwerking, associatief leren, emotionele filters, neurochemie en een hele reeks andere dingen die stuk voor stuk

ons denken beïnvloeden. KI kan veel sneller meer gegevens verwerken dan wij, maar doet dit niet op dezelfde manier. Menselijke intelligentie heeft jaren nodig om zelfbewust te worden, emotioneel begrip te ontwikkelen en de creatieve sprongen te maken die kunnen resulteren in iets artistieks. KI zal zich niet op dezelfde manier ontwikkelen wanneer we het gewoon tijd of meer rekenkracht geven. Wij geven KI vorm.

Het draait allemaal om jouw

Datamining wordt gebruikt om betere modellen te bouwen die zich zouden kunnen gedragen zoals wij. Met ChatGPT [2] ligt het misschien niet zo voor de hand, maar deze KI is ook op jou getraind. Op ons allemaal. Grote hoeveelheden tekst en onze tekstuele oprispingen en reacties vanuit het hele internet worden gebruikt om het taalmodel te trainen dat de kern vormt van ChatGPT. Als je meer wilt weten over deze onderliggende taalmodellen, raad ik je aan te luisteren naar "Speaking of Intelligence" van Hannah Fry [3] – een aflevering van haar DeepMind- podcastserie.

Een van de problemen van het model is dat het onbewust racistisch, agressief en ronduit afschuwelijk is. Het is getraind via het internet en heeft geen ingebouwd waardensysteem om de inhoud te beoordelen. Hoe verwerft het zulke waarden? Van mensen. Onderbetaalde computerkoelies die hun dagen doorbrengen met het bekijken van immorele teksten en afbeeldingen en het dienovereenkomstig labelen daarvan. Zodat het model wellicht leert om slechte inhoud te herkennen en deze niet meer te produceren [4]. Het modereren van sociale media-inhoud verloopt via eenzelfde proces. Dat het model steeds beter wordt, betekent dat mensen urenlang inhoud correct labelen als 'verkrachting', 'racistische inhoud', 'kindermisbruik', 'geweld' enzovoort. Veel van de intelligentie in KI is van onszelf.

ChatGPT is ook vaak feitelijk onjuist. Stack Overflow heeft het gebruik van ChatGPT voorlopig verboden omdat "het primaire probleem is dat, hoewel de antwoorden die ChatGPT produceert vaak onjuist zijn, ze doorgaans een plausibele indruk maken en ze heel gemakkelijk te produceren zijn" [5]. Omdat de KI op conversationele manier informatie levert, zijn we geneigd die informatie lichthartiger te verwerken en gemakkelijker voor waar te accepteren – als we er al een oordeel over vellen. Voordat je de volgende keer met ChatGPT in gesprek komt, wil ik je vragen om in gedachten te houden dat je tegen de mediaan van het internet praat, en het dienovereenkomstig te vertrouwen.

KI heeft BI nodig

We zouden de angel uit deze voortschrijdende arbeidsautomatisering kunnen halen door iedereen een onvoorwaardelijk basisinkomen te geven. Geef iedereen gewoon genoeg geld om van te leven. Geen kleine letterjes en geen vragen.

Generatieve KI's vormen geen bedreiging voor de creativiteit van een artiest; het bedreigt de manier waarom hij of zij een inkomen verdient. Ik zou KI kunnen gebruiken om afbeeldingen te maken die bij dit artikel passen in plaats van mijn gebruikelijke licentievrije stockfoto's of public domain-afbeeldingen, maar ik zou het nooit door een KI laten schrijven. Ik wil *zelf* schrijven. Het kan echter zijn dat mijn uitgever besluit dat het voordeliger is om een KI opdracht te geven artikelen over ethiek te schrijven in plaats van mij hiervoor te betalen. Dat doet er opeens een stuk minder toe als mijn huis en gezondheid niet van die betaling afhankelijk zijn.

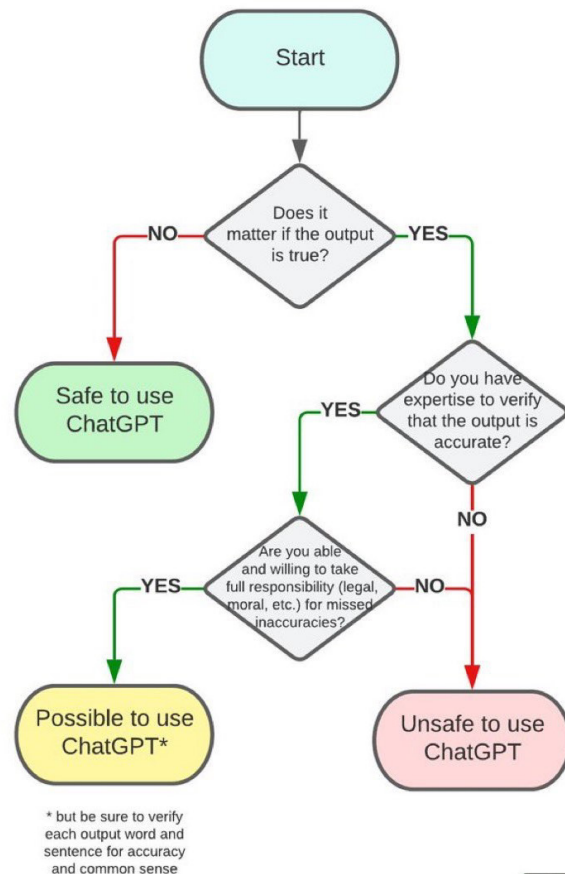
Vergis je niet: geen beroep is veilig voor KI. Een deel van het werk van beeldend kunstenaars, schrijvers, advocaten, programmeurs en ingenieurs zal in de nabije toekomst ongetwijfeld worden overgenomen door generatieve KI. We kunnen KI misschien als hulpmiddel gebruiken terwijl we onszelf concentreren op wat we het liefste doen als ons levensonderhoud is veiliggesteld. Een onvoorwaardelijk basisinkomen kan ons de veiligheid bieden om te trachten het werk te verdelen tussen wat mensen het beste doen en wat KI beter kan.

Wanneer we KI als een nuttig hulpmiddel kunnen beschouwen, wordt duidelijk dat wij mensen de beslissingen moeten blijven nemen. De beslissing om de uitvoer van de KI al dan toe te passen. Misschien de beslissing om de prompt anders te formuleren of de parameters te wijzigen en de KI een nieuwe poging te laten doen. Beslissen wat we doen met de uitvoer van de KI maakt deze ook tot onze verantwoordelijkheid. In de auto, in een medische situatie en bij het delen van informatie met anderen. ◀

230048-01

Is it safe to use ChatGPT for your task?

Aleksandr Tiulkanov | January 19, 2023



ChatGPT-stroomschema. Aleksandr Tiulkanov is KI-onderzoeker.

Over de auteur

Priscilla Haring-Kuipers schrijft over technologie vanuit sociaal-wetenschappelijk perspectief. Ze is vooral geïnteresseerd in technologie die het goede in de mensheid ondersteunt en gelooft heilig in effect-onderzoek. Ze heeft een MSc in mediapsychologie en is een van de mensen achter This Is Not Rocket Science.

Het World Ethical Electronics Forum

Het World Ethical Electronics Forum (WEEF) inspireert innovators wereldwijd met open discussies en publicaties over ethiek en doelstellingen voor duurzame ontwikkeling. Bezoek worldethicalelectronicsforum.com voor inspiratie en deelname.



WEBLINKS

- [1] OpenAI, DALL-E 2: <https://openai.com/dall-e-2/>
- [2] OpenAI, "ChatGPT: Optimizing Language Models for Dialogue", 30 november 2022: <https://openai.com/blog/chatgpt/>
- [3] H. Fry, "Speaking of intelligence", DeepMind: The Podcast (S2, Ep2), 25 januari 2022: <https://youtu.be/21JSKHR7KWw>
- [4] B. Perrigo, "OpenAI Used Kenyan Workers on Less Than \$2 Per Hour to Make ChatGPT Less Toxic", 18 januari 2023: <https://time.com/6247678/openai-chatgpt-kenya-workers/>
- [5] Stack Overflow, "Temporary Policy: ChatGPT Is Banned", 5 december 2022: <https://meta.stackoverflow.com/questions/421831/temporary-policy-chatgpt-is-banned>

De Elektor Store

nooit duur, altijd verrassend!

De Elektor-store heeft zich ontwikkeld van de community-shop voor de eigen producten van Elektor (boeken, tijdschriften, kits en modules) tot een volwassen webshop die veel waardevolle elektronica-aanbiedingen heeft.

We bieden hier producten aan waar we zelf enthousiast over zijn of die we gewoon willen uitproberen. Suggesties zijn altijd welkom (sale@elektor.nl).

Elektor Archief 1980-2022 (USB-stick)

4 Elektor decennia (jaren '80, '90, '2000 en '2010) op USB-stick

Deze USB-stick (32 GB, USB 3.0) bevat de complete jaargangen 1980-2022 (alle nummers) van het elektronica-vakblad Elektor. Elektor wil mensen inspireren om zich elektronica en computertechniek eigen te maken door het presenteren van nabouwwriendelijke, professioneel ontworpen schakelingen op alle terreinen van de elektronica.

Prijs: ~~€ 199,95~~

Actieprijs: € 99,95

 www.elektor.nl/20374

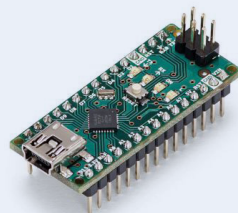
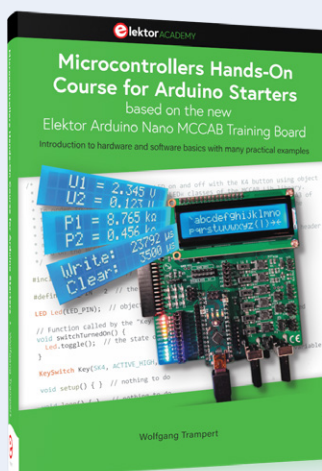


Microcontrollers Hands-on Course for Arduino Starters (Bundel)

Deze praktische cursus microcontrollers voor Arduino starters laat zien hoe u uw eigen projecten kunt realiseren met een microcontroller, zelfs zonder veel ervaring te hebben met elektronica en programmeertalen.

Prijs: ~~€ 109,95~~

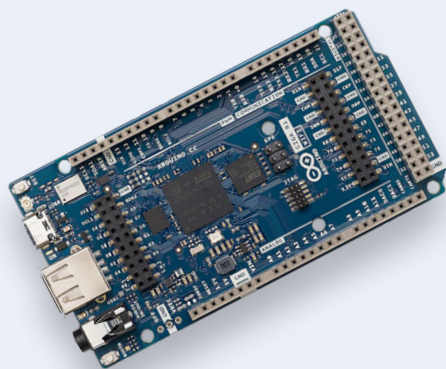
Actieprijs: € 94,95



 www.elektor.nl/20440



Arduino Giga R1 WiFi

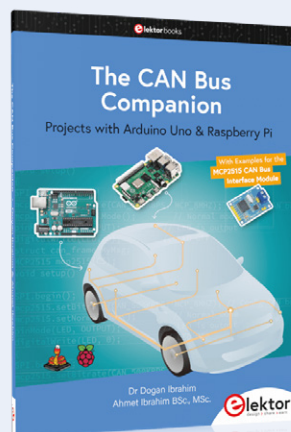


Prijs: € 89,95

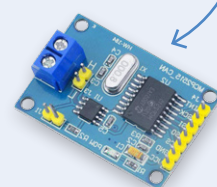
Ledenprijs: € 80,96

www.elektor.nl/20436

The CAN Bus Companion



+
GRATIS CAN-module



Prijs: € ~~37,95~~

Actieprijs: € 29,95

www.elektor.nl/20405

SDRplay RSP1A – 14-bit SDR-receiver (1 kHz tot 2 GHz)



Prijs: € 139,95

Ledenprijs: € 125,96

www.elektor.nl/20421

Whadda WTS100 Soldeer starterset



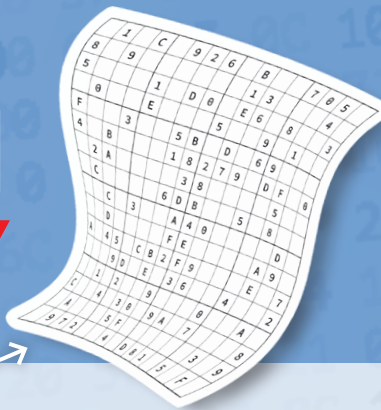
Prijs: € 49,95

Ledenprijs: € 44,96

www.elektor.nl/20426

Hexadoku

puzzelen voor elektronici



PC, oscilloscoop en soldeerbout kunnen weer even op adem komen terwijl u uw hersenen pijnigt met onze Hexadoku. De instructies voor deze puzzel zijn heel eenvoudig. De Hexadoku werkt met de hexadecimale cijfers 0 t/m F, helemaal in de stijl van elektronici en programmeurs.

Vul het diagram van 16 x 16 hokjes zodanig in dat **alle** hexadecimale cijfers van 0 t/m F (dus 0...9 en A...F) precies éénmaal voorkomen in elke rij, in elke kolom en in elk vak van 4 x 4 hokjes (gemarkeerd door de dikkere zwarte lijnen). Een aantal cijfers is al aangegeven en deze bepalen de uitgangssituatie voor de puzzel.

Onder de inzenders met de goede oplossing verloten we vijf waardebonnen. Om mee te dingen naar een van deze prijzen dient u **de cijfers in de grijze hokjes** naar ons op te sturen.



Doe mee en win!

Onder de internationale inzenders met het juiste antwoord verloten we vijf Elektor-waardebonnen, elk ter waarde van **50 Euro**.

Het is dus zeker de moeite waard om mee te doen!

Stuur uw antwoord (de getallen in de grijze hokjes) vóór **15 juni 2023** naar:

hexadoku@elektor.nl

DE PRIJSWINNAARS

De juiste oplossing van de Hexadoku uit het maart/april-nummer 2023 is: **F13AB**.

Oplossingen die ons vóór 15 april 2022 hebben bereikt, deden mee aan de trekking van vijf Elektor-waardebonnen.

De winnaars zijn bekendgemaakt op www.elektormagazine.com/hexadoku.

Allemaal van harte gefeliciteerd!

E	2		A	0			B		F		C	6	4	9	
	C		F	1	5	8					0		B	A	7
		1				4							8		2
8	3	4				C	D		7	B	E		0	1	
		F	8								3			2	9
7		2	3			1				9	B	D			A
	6				D		8		A	E		5			
	1			5			6		8		D				
	7			A			3		E		6				
	D				6		2		B	3		8			
3		9	6			5				7	F	1			0
		0	5								9			7	6
0	9	E				2	A		3	F	5		C	6	
		7				6							5		3
	8		C	F	7	3					1		D	0	B
5	F		4	8			0		6		2	A	7	E	

A	1	3	F	B	0	C	D	6	4	8	5	E	2	7	9
0	6	5	2	9	E	F	1	3	A	B	7	4	8	C	D
4	8	C	7	5	2	A	3	F	9	E	D	0	1	6	B
9	B	D	E	4	6	7	8	1	C	0	2	F	3	A	5
F	7	E	8	0	3	D	9	2	B	4	C	A	6	5	1
2	9	0	A	6	1	8	7	5	E	3	F	D	4	B	C
B	D	4	5	C	F	E	A	0	1	9	6	2	7	8	3
1	C	6	3	2	4	5	B	D	7	A	8	9	E	0	F
3	4	1	6	F	8	0	C	7	D	2	B	5	9	E	A
C	E	A	B	7	D	9	2	8	5	F	3	6	0	1	4
5	F	7	D	1	A	6	E	4	0	C	9	3	B	2	8
8	0	2	9	3	5	B	4	A	6	1	E	C	D	F	7
D	2	F	4	A	B	1	6	9	8	5	0	7	C	3	E
E	3	9	C	8	7	2	0	B	F	D	A	1	5	4	6
6	A	B	1	E	9	3	5	C	2	7	4	8	F	D	0
7	5	8	0	D	C	4	F	E	3	6	1	B	A	9	2

Medewerkers van Elektor International Media en hun familieleden zijn van deelname uitgesloten.

Word lid van de Elektor Community



Neem **nu** een
lidmaatschap!



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



Ook verkrijgbaar

Het digitale
lidmaatschap!



- ✓ Toegang tot ons web-archief
- ✓ 10% korting in onze webshop
- ✓ 8x Elektor Magazine (PDF)
- ✓ Toegang tot meer dan 5000 Gerberfiles



www.elektormagazine.nl/Abonnement

Ontwikkelingstools op één plaats

Duizenden tools van honderden
vertrouwde fabrikanten



Maak een keuze uit onze
uitgebreide selectie op
nl.mouser.com/dev-tools



MOUSER
ELECTRONICS