januari/februari 2019 | Nr. 651 | www.elektormagazine.nl | € 14,95

elektor labs magazine

DESIGN ⊢ BUILD ⊢ SHARE ELECTRONICS

ESP32-Weerstation

Sensordata online uitlezen

Zelfbouw-soldeerstation met temperatuurregeling

SMD-solderen voor iedereen

FM-radio met RDS

Een hat voor de Raspberry Pi

Elekromotoren aansturen ⊢ Snore Shield ⊢ Vuurtorenlicht 2.0 ⊢ One-time Pad Crypto Shield ⊢ Halfduplex-telemetrie op 2,4 GHz ⊢ Retrotronica: Netspanning filteren en stabiliseren, zo deed men dat vroeger ⊢ Verilog Basics ⊢ Speed-o-meter ⊢ PobDuino Board ⊢ Opamp Schmitt Triggers ⊢ V&A: Behuizingen ⊢ Instelbare frequentiedeler ⊢ Ethics: Het Interplanetary File System

BP 00651
8 710966 982907

WORD ABONNEE EN ONTVANG EEN

GRATIS PI ZERO W

SLECHTS € 54,95 PER JAAR (6 NUMMERS)

IEDERE 2 MAANDEN HET LAATSTE RASPBERRY PI NIEUWS EN DE ALLERLEUKSTE PROJECTEN!

Neem nu een jaarabonnement op MagPi en ontvang:

- Zes edities van MagPi Magazine
- Gratis Raspberry Pi Zero W
- Gratis Pi Zero W case met 3 covers
- Gratis camera module connector
- Gratis converterkabels voor USB en HDMI

Uw voordelen:

- Goedkoper dan zes losse nummers
- Iedere editie in de brievenbus; u hoeft de deur niet uit
- Elk nummer ook digitaal beschikbaar (PDF)
- Exclusief welkomstcadeau t.w.v. € 22,95
- U leest MagPi al voordat het blad in de winkel ligt

ABONNEER NU OP WWW.MAGPI.NL

Colofon

59e jaargang nr. 651,
januari/februari 2019
ISSN 2590-0765

Elektorlabs is een uitgave van
Elektor International Media B.V.
Postbus 11, 6114 ZG Susteren, Nederland
Tel.: +31 (0)46- 4389444,
Fax: +31 (0)46-4370161

Nieuwe abonnementen & bestellingen
service@elektor.nl Tel. 046-4389444

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

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

Of per email: service@elektor.nl

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

Druk: Senefelder Misset, Doetinchem
Distributie: Betapress, Gilze

Advertenties Benelux

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

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

Auteursrecht
Niets uit deze uitgave mag verveelvoudigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm of op welke wijze dan ook, zonder voorafgaande schriftelijke toestemming van de uitgever. De auteursrechtelijke bescherming van Elektor strekt zich mede uit tot de illustraties met inbegrip van de printed circuits, evenals de ontwerpen daarvoor. In verband met artikel 30 van de 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.

© Elektor International Media B.V. - 2019

Terug van weggeweest...

Zo'n anderhalf jaar geleden werd mij verzocht de Nederlandse editie van Elektor onder mijn hoede te nemen. Een hele eer natuurlijk (ondanks de wat minder prettige aanleiding — ziekte van *editor in chief* Thijs Beckers) en een hele verantwoording... In die anderhalf jaar heb ik het voorrecht gehad in een fijn team te werken, en heb ik veel geleerd. Vooral dat ondanks de voortschrijdende digitalisering en computerisering (die ik niet altijd als een onverdeelde zegen beschouw) het maken van het maken van het blad dat u nu in uw handen houdt veel hard werk is — mensenwerk wel te verstaan. En mensen maken fouten en foutjes — een fraai voorbeeld treft u aan op bladzijde 82.
Hoe het ook zij — met groot genoegen kan ik u meedelen dat Thijs na lange afwezigheid weer terug is van weggeweest, en dat hij vanaf het maart/april-nummer van Elektorlabs weer de touwtjes in zijn vaardige handen heeft. En ik ben — om het met een knipoog te zeggen — weg van teruggeweest (maar blijf op de achtergrond actief).
Eric Bogers

Wij zijn Eric uiteraard zeer erkentelijk dat hij ons lijfblad de afgelopen tijd onder zijn hoede heeft genomen.
Bedankt, Eric!
Terug in het zadel... het is nog wel wat onwennig. Er is in de tussentijd toch wel het een en ander veranderd aan de gang van zaken binnen Elektor. Andere collega's, andere aanspreekpunten... En zoals Eric ook al opmerkte, is de digitale opmars ook bij ons gedrukte blaadje geen halt toe te roepen. Achter de schermen is nu alles gedigitaliseerd en online bereikbaar, en kan het blad in theorie vanaf het Sunset Beach in Hawaï met een cocktail in de hand op een laptop in elkaar gezet worden. Nou zie ik het nog niet gebeuren dat we dit in de praktijk op die manier ook mogen gaan testen, maar anders geef ik mij graag op als vrijwilliger. Genoeg gekheid! We ontwerpen onze schakelingen en maken ons blad met de grootst mogelijke aandacht, zodat u er van kunt genieten. Vóór u ligt weer een nieuwe editie die vol staat met projecten, ideeën en achtergrondinformatie. Papier heeft zijn aantrekkingskracht nog lang niet verloren en ik moet zeggen dat ik zelf ook liever lees van papier dan van een beeldscherm. Veel lees- en bouwplezier!
Thijs Beckers

Ons team

Hoofdredacteur:	Thijs Beckers (thijs.beckers@eimworld.com)
Internationale redactie:	Jan Buiting, Jens Nickel, Mariline Thiebaut-Brodier
Technische redactie:	Mathias Claussen, Ton Giesberts, Luc Lemmens, Denis Meyer, Clemens Valens, Jan Visser
Aan dit nummer werkten mee:	Jelle Aarnoudse, Rolf Blijleven, Eric Bogers, Kees de Groot, Martien Jansen, Evelien Snel
Ledenmanagement:	Raoul Morreau (raoul.morreau@eimworld.com)
Vormgeving & Layout:	Giel Dols, Mart Schroijen, Patrick Wielders
Hoofd Online:	Daniëlle Mertens
Directeur:	Don Akkermans

in dit nummer

59e jaargang – nr. 651

januari/februari 2019

20 electronica Fast-Forward 2018
een stukje toekomstvisie van 'smart'-start-ups

22 Elektor Connexions

23 Escaped from the Labs
Obsolete of 'allocated'?

33 Vreemde onderdelen
de SN76477

34 HomeLab Helicopter

38, 99 Tips en trucs
van lezers voor lezers

61 Elektor Labs Pipeline

80 V & A
behuizingen

105 Shop-highlight
waar elektronici van dromen...

106 Retrotronica
netspanning filteren en stabiliseren, zo deed men dat vroeger

112 Elektor Store

114 Hexadoku

achtergrond & info

6 Elektromotoren aansturen
met een schakelaar, of een eenvoudige PWM, of een brugschakeling...

62 Schmitt-triggers met opamps
schakelingen en berekeningen

74 Nieuw bij Android Things
C++ als booster

82 Verkeerde fout in de schakeling!
hij doet het wel, maar anders dan gedacht ;-)

94 Kristallen
slechts twee pootjes, maar erg interessante eigenschappen!

100 Verilog Basics
beginselen van de hardware-beschrijvingstaal Verilog

110 Elektor Ethiek
Het Interplanetary File System

14

Zelfbouw-soldeerstation met temperatuurregeling

SMD-solderen voor iedereen

Het Platino-soldeerstation dat in 2015 werd gepubliceerd, kreeg brede bijval — en niet alleen van het lezerspubliek van Elektor. Alleen al het feit dat het oorspronkelijke project meteen aansloeg, was in de beste ontwerptraditie een belangrijke reden om het oorspronkelijke ontwerp nog eens goed tegen het licht te houden. Alle hieraan bestede inspanningen en ideeën hebben een volledig nieuw ontwerp opgeleverd dat we hier zullen beschrijven.

projecten

13 Instelbare frequentiedeler
met een minimum aan (CMOS-) onderdelen

14 Zelfbouw-soldeerstation met temperatuurregeling
SMD-solderen voor iedereen

24 Vuurtorenlicht 2.0
met instelbaar lichtkarakter

6

Elektromotoren aansturen

met schakelaar, eenvoudige PWM, of brugschakeling ...

elektorlabs magazine

56 Halfduplex-telemetrie op 2,4 GHz
goedkoop en eenvoudig
met kant-en-klare modules

68 PobDuino
Flowcode en Arduino geven oude robots
een nieuw leven

86 FM-radio met RDS
een hoed voor de Raspberry Pi

FM-radio met RDS

een hoed voor de Raspberry Pi

Een Raspberry Pi, een zelf ontwikkelde HAT met stuurprogramma's en een grafische gebruikersinterface die speciaal voor deze toepassing zijn geschreven, zijn de ingrediënten van een VHF-radio die er niet alleen interessant uitziet, maar ook uiterst instructief is dankzij de uitgebreide documentatie. Omdat je nooit uitgeleerd raakt!

86

binnenkort

Elektorlabs magazine maart/april 2019

3-assige CNC-machine • 3-kanaals batterijmonitor • ultrasone fietsradar • baristor • audio-spectrummeter • neuraal netwerk met Arduino • microcontroller-kits voor dummies • doe-het-zelf CPU in FPGA • VHDL-cursus deel 3 • stappenmotoren aansturen zonder resonanties • Qi-laders • retrotronica • AD-omzetter met PLD • en nog veel meer.

Aankondigingen onder voorbehoud.
Elektorlabs maart/april 2019 verschijnt op 21 februari 2019

Elektor Industry 1/2019

Elektor Industry 1/2019 behandelt microcontrollers, embedded technologie, programmeren en tools, met bijdragen van bedrijven, specialisten uit de industrie, redacteuren van Elektor en zelfstandige auteurs. Natuurlijk treft u ook nieuwe afleveringen aan van vaste rubrieken zoals Infographics en Industry Store.
Nummer 1/2019 besteedt speciale aandacht aan de **Embedded World Exhibition & Conference** in Neurenberg (Duitsland) van 26 tot 29 februari 2019, waar Elektor ook vertegenwoordigd zal zijn.

Elektor Industry 1/2019 verschijnt op 20 februari 2019 in druk voor Elektor Gold-leden en als pdf-download voor Elektor Green-leden. Het nummer is voor belangstellenden ook te koop via www.elektormagazine.com. Aankondigingen onder voorbehoud.

28 ESP32-weerstation
sensordata online uitlezen

40 Snore Shield
helpt je nachtrust te verbeteren

45 Speed-o-meter
met tripmeter en logger

48 One-time Pad Crypto Shield
veilige communicatie via een open systeem

Elektromotoren aansturen

met een schakelaar, of een eenvoudige PWM, of een brugschakeling…

Dr. Thomas Scherer (Duitsland)

Overal waar stroom in kracht en elektrische energie in (doorgaans roterende) mechanische energie wordt omgezet, hebben we te maken met de goede oude elektromotor. 'Goed' omdat er bij dit soort motoren veel minder mechaniek 'stoort' dan bijvoorbeeld bij een stoommachine of een verbrandingsmotor — met dank aan de elektrotechnici die ons voorgingen. 'Oud' voor elektronici omdat eigenlijk alle elektromotoren al lang uitgevonden zijn en de technologie nauwelijks nog verandert. Maar er zijn allerlei soorten elektromotoren die voor specifieke doeleinden op de juiste manier moeten worden aangestuurd.

Bij thermomechanische motoren is er altijd iets heets dat (met veel lawaai, rook en stank) via een krukas of iets dergelijks iets anders in beweging brengt. Elektromotoren, die in talloze uitvoeringen bestaan, brengen daarentegen bijna altijd de rotatie-as direct in beweging. Dat maakt niet alleen de mechanica eenvoudiger, maar zorgt ook voor een lager gewicht en lagere kosten bij de toepassing. Dankzij die eenvoud gaan elektromotoren ook langer mee, en is het eenvoudiger ze nauwkeurig aan te sturen. Tegenover deze voordelen staat het nadeel dat er aan een elektromotor nagenoeg altijd een kabel hangt, en anders wel een accu die de zaak echter weer zwaarder, duurder en omslachtiger maakt.

Figuur 1. Jedlik-Motor uit 1827 (bron: Wikimedia Commons [2]).

Figuur 2. De legendarische Lohner-Porsche, foto genomen rond 1900 met rechts Ferdinand Porsche. Moderne aandrijving met vier naafmotoren van elk 1,5 kW (bron: Michael Hereward Westbrook in „The Electric Car" [3]).

Figuur 3. Al in 1885 toonde het 'Meyersche Konversations-Lexikon' (4e druk) allerlei soorten constructies van elektromotoren [4][5].

Verscheidenheid

Toen de Deen Hans-Christian Ørsted zo'n 200 jaar geleden merkte dat een stroomvoerende geleider een kompasnaald beïnvloedt, ontketende hij daarmee het vakgebied elektrotechniek in het algemeen en de stormachtige ontwikkeling van de elektromotor in het bijzonder (**figuur 1**). Amper twintig jaar later voer er in het Russische Sint Petersburg al een elektrisch aangedreven boot op de Neva. En iedere *Homo technicus* weet dat er in de begindagen van de automobiel meer met elektro- dan met benzinemotoren rondreden (**figuur 2**). Maar pas na de aanleg van elektrische distributienetten in de Verenigde Staten door Edison resp. General Electric en Tesla resp. Westinghouse, en in het toenmalige Duitse rijk door AEG, werden elektromotoren op grote schaal voor de meest uiteenlopende toepassingen ingezet. Paarden-, water- en windkracht werden snel van ondergeschikt belang en er ontsloten zich mogelijkheden om dingen te motoriseren waar men voorheen alleen maar van kon dromen.

Sinds het eind van de negentiende eeuw hebben allerlei knutselaars en ingenieurs zich flink uitgeleefd met elektrische aan-

Tabel 1. Typen elektromotoren.

Benaming	Vermogen	Kenmerken en bijzonderheden
Asynchrone draaistroommotor	W - MW	Borstellloos (geen collector), de rotor draait trager dan het statorveld, aansturing via schakelaars of frequentie-omzetters (toerental), hoge aanloopstroom (ster/driehoek-omschakeling).
Synchrone draaistroommotor	W - MW	Aanloopveld noodzakelijk (permanent of elektrisch), rotor draait synchroon met het draaiveld, frequentieregelaar (toerental), zelfsturend draaiveld via elektronica, generator, binnen- of buitenpoolmachine.
Stappenmotor	<1 W - 1 kW	Variant van de synchroonmotor met stapsgewijze rotordraaiing over een bepaalde hoek, nauwkeurige positionering, 24 tot 400 stappen per omwenteling.
BLDC-motor	W - MW	Borstelloze gelijkstroommotor (*brushless DC)*, variant van de synchroonmotor met elektronische commutatie, voor aandrijvingen, zeer betrouwbaar, vaak met permanente magneten.
Condensator-motor	W - <1 kW	Eénfasige asynchroon- of synchroonmotor, draaiveld d.m.v. faseverschuiving in de aanloopwikkeling, gering aanloopkoppel, onderhoudsarm, eenvoudig.
Spleetpoolmotor	W - <1 kW	Eénfasige asynchroon- of synchroonmotor, draaiveld d.m.v. kortsluitwikkeling in de spleetpool, onderhoudsarm, gering aanloopkoppel, geluidsarm, gering vermogen.
Gelijkstroommotor	W - kW	Permanent of elektrisch opgestart, mechanische commutator maakt het wisselveld, zeer veel varianten, serie- of shuntmotor, hoog aanloopkoppel.
Eénfase seriemotor	W - kW	Als een gelijkstroommotor met blikpakket tegen wervelstroomverliezen bij wisselstroomvoeding, hoog toerental, borstelslijtage, EMV, hoog aanloopkoppel.
Lineaire motor	W - kW	Geen rotor, lineaire beweging, hoog koppel, veel varianten, positionering, harddisks, monorail.
Servo	W	Combinatie van elektromotor met aansturing en referentiesignaal voor de draaihoek, modelbouw, PWM-aansturing, drie aansluitingen.

Figuur 4. Principe en opbouw van een simpele elektromotor met permanente magneet. De bovenkant van de stator is de noordpool, de onderkant is de zuidpool. De rotor is een elektromagneet die gevoed wordt via een commutator bestaande uit koolborstels en koperen sleepringen. (Tekening naar Michael Frey, CC 3.0 [6]).

drijving en zijn de meest uiteenlopende soorten elektromotoren ontwikkeld. Zegt het begrip 'stroomoorlog' u iets [1]? GE en Westinghouse, opgericht door de grote uitvinders Edison en Tesla, stonden lijnrecht tegenover elkaar met betrekking tot de principiële vraag: gelijkstroom of wisselstroom? Ook toen al had je niet alleen elektromotoren met de meest uiteenlopende prestaties en constructie (**figuur 3**), maar ook motoren die ontworpen waren voor gelijk- c.q. wisselstroom, en in het laatste geval met verschillende aantallen fasen waren uitgerust. Voor iedere toepassing was er een speciale motor. Deze veelheid aan aanduidingen en de daarmee samenhangende verschillen kunnen verwarrend zijn voor de elektronicus die hierin niet gespecialiseerd is. Daarom hebben we in **tabel 1** de belangrijkste typen elektromotoren met hun eigenschappen voor u opgesomd.

Principes

In dit artikel richten we ons op hoe men elektromotoren toepast en hoe die worden aangestuurd. We gaan daarom niet diep in op de specifieke werking van elk afzonderlijk motortype. In principe is een elektromotor immers een constructie van één of meer of zelfs veel elektromagneten waarvan het magnetisch veld een kracht uitoefent op een andere magneet of op een permanente magneet of op een speciaal gevormd stuk ijzer; die kracht veroorzaakt een mechanische beweging. Elektrische energie wordt daarbij omgezet in mechanische energie. Zoals te verwachten gebeurt die omzetting niet volledig, niet alleen omdat een deel van de energie verloren gaat door mechanische effecten zoals wrijving, maar ook doordat er elektrische en elektromechanische effecten aan het rendement afbreuk doen. Als eerste noemen we hier de ohmse weerstand die tot warmteontwikkeling leidt in plaats van tot bewegingsenergie. Bij hogerfrequente wisselstromen wordt die warmteproductie nog eens vergroot door het zogenaamde skineffect. Bij gelijkstroom is daar niets van te merken, bij de gebruikelijke 50 of 60 Hz is het effect zeer bescheiden en nauwelijks problematisch, maar al bij zo'n 30 kHz neemt de zogenaamde indringdiepte van de wisselstroom af tot zo'n 0,4 mm. Daarbij komen nog verschijnselen als wervelstroomverliezen in de ijzerhoudende delen van de motor die door veranderlijke magneetvelden worden doorsneden. En tenslotte zijn er secundaire oorzaken zoals wat er zich afspeelt tussen sleepringen en koolborstels en dergelijke, voor zover de motor daarmee is uitgerust.

En *last but not least* hebben we te maken met verliezen in de aansturing. Bij slechte elektromechanische schakelaars en relais zijn die terug te voeren op verliezen in de relaisspoelen en op de meestal zeer lage contact- en kabelweerstanden. Bij aansturing met halfgeleiders hebben we te maken met het energieverbruik van stuurelektronica en (in mindere mate) netvoeding, en verder met de spanningsval over de geleidende schakelelementen, plus het energieverlies bij snel schakelen — wat bij PWM-aansturing eigenlijk altijd gebeurt.

Moderne power-MOSFET's zijn in dit opzicht voor kleine en gemiddelde motorvermogens (tot enkele kW) prima geschikt omdat hun on-weerstand in de orde van grootte van enkele mΩ kan liggen. Bij grotere vermogens, zoals in auto's en de daarmee gepaard gaande hogere spanningen neemt de on-kanaalweerstand van silicium-MOSFET's echter aanzienlijk toe. Dit heeft in de afgelopen jaren geleid tot de opkomst van vermogenshalfgeleiders van siliciumcarbide. Met dit materiaal zijn hogere veldsterktes mogelijk en dus dunnere lagen. SiC-MOSFET's zijn daardoor bij uitstek geschikt voor hogere spanningen en maken hogere schakelfrequenties en lagere on-weerstanden mogelijk, met lagere verliezen als gevolg. Bovendien zijn SiC-transistoren te gebruiken bij hogere temperaturen, waardoor de betrouwbaarheid toeneemt. Maar zoals zo vaak in dit leven hangt er aan deze verbeteringen wel een hoger prijskaartje.

Werking

Zoals gezegd zou het te ver voeren om van elk type motor in detail te bespreken hoe die concreet werkt. Niettemin helpt het wel om de basisprincipes van de elektromotor in het achterhoofd te houden als we ingaan op het gebruik en de aansturing. **Figuur 4** toont het principe van een (vereenvoudigde) elektromotor. Aan de buitenkant resp. in de figuur boven (noord) en onder (zuid) in beeld zien we de polen van een permanente magneet. Bij dit type motor beweegt die magneet niet, daarom heet hij stator. Het deel tussen de beide polen draait wel, en heet bijgevolg rotor of ook wel anker. De rotor bestaat uit een ijzeren kern met een wikkeling van geïsoleerd koperdraad. De rotor fungeert als een gelagerde, draaibare elektromagneet. In deze uitvoering heet deze constructie een *binnenloper.* U kunt zich voorstellen dat de constructie ook andersom mogelijk is, met een stilstaand anker en een gelagerd draaibaar buitendeel. Zo zitten bijvoorbeeld naafmotoren in elkaar. Die noemen we *buitenlopers*.

Als we nu op de een of andere manier gelijkstroom door de wikkeling laten lopen, dan willen de blauwe noordpool van de rotor en de rode zuidpool van de stator naar elkaar toe, net als de zuidpool van de rotor en de noordpool van de stator. Dit gebeurt heel snel; de rotor is dan (in de figuur) ongeveer 60° gedraaid en blijft daar dan als vastgenageld staan.

Dat zou vrij nutteloos zijn. Daarom wordt de stroom via borstels (zwart in de tekening, meestal van koolstof — vandaar de naam koolborstels — en bij de allerkleinste motoren ook wel contactveren van edelmetaal) en twee losse helften van een sleepring (oranje in de tekening) naar de ankerwikkeling geleid. Deze constructie heet commutator en poolt de stroom

Figuur 5. Deze vermogensthyristor voor 4,2 kV bij 2,4 kA heeft een diameter van 11 cm. (Bron: Johnny.m76 / CC 3.0 [7]).

Figuur 6. Omkeren van de draairichting van een eenvoudige gelijkstroommotor gebeurt met twee gekoppelde schakelaars.

door de wikkeling om al naargelang de positie van de rotor, zodanig dat de stroomrichting omkeert zodra de rotor rechtop komt. Dat zorgt voor een continu draaimoment in één richting. Dit principe is even eenvoudig als geniaal, en er zijn vele varianten op denkbaar. Niet alleen kunnen we binnen en buiten omwisselen, maar ook de permanente magneet en de elektromagneet. We kunnen ook de permanente magneet vervangen door een tweede elektromagneet. In het voorbeeld van figuur 4 krijgt de stator dan ook een wikkeling. En zo ontstaan nog talloze andere mogelijkheden: de tweede wikkeling kan in serie met of parallel aan de eerste worden geschakeld. In het eerste geval krijgen we dan een seriemotor, niet geheel ten onrechte aangeduid als 'universele motor'. Deze variant komen we werkelijk overal tegen, van boormachines via stofzuigers en wasmachines tot industriële toepassingen. En hij wordt universeel genoemd omdat hij met zowel wissel- als gelijkstroom bruikbaar is.

Een nadeel van commutatoren is slijtage. De koolborstels slijten, en op de lange termijn zijn ook de contact-oppervlakken van de sleepringen 'op'. Om die reden worden bijvoorbeeld meerfase-motoren gemaakt: aangesloten aan een meerfasig net benutten ze het draaiveld en zijn sleepringen niet nodig. Er bestaan nog meer ingenieuze constructies waarbij ook met een enkelfasige aansluiting een draaiveld gemaakt wordt met speciale kortsluitwikkelingen. Verder kunnen we de functie van de sleepringen tegenwoordig ook laten vervullen door slijtvaste elektronische schakelaars, die worden aangestuurd door positie-sensoren in de motor zelf of door de tegen-EMK. Het toerental van synchroonmotoren die op de netspanning zijn aangesloten, kan binnen zekeregrenzen met omvormers worden gevarieerd. Al deze variaties brengen een veelheid aan soorten elektromotoren met zich mee. In **Tabel 1** hebben we de belangrijkste categorieën voor u op een rijtje gezet.

Ook goed om te weten: bijna alle motortypen kunnen ook fungeren als generator, en daarmee elektrische energie opwekken uit bewegingsenergie. Soms zijn hier slechts kleine aanpassingen in de constructie voor nodig, en soms kan hier nuttig gebruik van worden gemaakt. Zo kunnen motoren bijvoorbeeld worden afgeremd door de kinetische energie na het uitschakelen te 'verstoken' (ook in de wikkeling). In andere gevallen kan het ook storend zijn als de generatorwerking van een uitgeschakelde maar nog draaiende motor de aangesloten stuurelektronica activeert.

Aansturingen

Met zoveel verschillende soorten elektromotoren zijn er net zoveel verschillende aansturingsmogelijkheden. Een eenvoudige schakelaar beantwoordt maar zelden aan de eisen en mogelijkheden van de diverse elektromotoren en dus ook even zelden aan de beoogde toepassingen. Vandaar dat we hieronder de meest voorkomende aansturingsmethodes bespreken, in volgorde van toenemende complexiteit. Modelbouw-servo's en stappenmotoren laten we buiten beschouwing omdat die voor aandrijving minder geschikt zijn en eigenlijk compleet anders zijn.

Schakelaars

Motoren met klein tot middelgroot aandrijfvermogen worden meestal gewoon geschakeld. In het eenvoudigste geval gebeurt dat met een mechanische schakelaar. In plaats daarvan kan een relais worden toegepast, zodat de motor via een (lange) draad op afstand schakelbaar is. Voor motoren die op de netspanning werken zijn er vermogensrelais. Naast deze elektromechanische schakelaars zijn er ook elektronische alternatieven in de vorm van halfgeleider-relais voor kleine en middelgrote vermogens, en kleine tot grote thyristoren en triacs (**figuur 5**), waarmee motoren met vermogens tot vele kW niet alleen geschakeld kunnen worden, maar waarmee ook het toerental en het koppel van de motor geregeld kan worden. Met schakelaars kan nog meer, zoals we hieronder zullen zien.

Draairichting via schakelaars

Een van de eenvoudigste mogelijkheden om de draairichting van een gelijkstroommotor om te keren is een dubbelpolige schakelaar (**figuur 6**). In principe is elk type schakelaar geschikt en het kan ook met halfgeleiders. Pas wel op, want u kunt niet zomaar een draaiende motor zonder pauze direct ompolen. Zou je dat wel doen, dan ontstaan er door de generatorwerking akelig grote stromen en spanningspieken, en kan het de mechanica van de motor of de ophanging worden beschadigd

Figuur 7. Het toerental van gelijkstroommotoren kan worden geregeld met een halve brug met twee elektronische schakelaars, aangestuurd met twee PWM-signalen die met elkaar in tegenfase zijn.

Figuur 8. Met een volledige of H-brug, eigenlijk een dubbele versie van figuur 7, zijn toerental en draairichting te regelen via PWM.

door het grote koppel dat er dan ontstaat. De gevolgen zijn erger dan wanneer de wikkelingen worden kortgesloten. Dat iets 'in principe' mogelijk is, wil echt niet zeggen dat het ook goed werkt of überhaupt uitvoerbaar is. Behalve bij kleine motortjes moet een mechanische draairichting-omschakelaar een midden- of een derde stand hebben waarin de motor eerst wordt afgeremd.

Er bestaan ook allerlei soorten enkelfasige wisselstroommotoren met twee gescheiden wikkelingen waarmee de draairichting veranderd kan worden. Daarbij hoeft u slechts één van beide wikkelingen om te polen. Bij veel draaistroommotoren kan men gewoon twee fasen omwisselen. Wel is het zo dat wisselstroommotoren meestal een voorkeurs-draairichting hebben. Bij dat type motor is de commutator enkele graden gedraaid ten opzichte van het theoretische origineel, wat onder andere een beter rendement kan opleveren.

Toerental via PWM

Bij gelijkstroommotoren is behalve de draairichting ook het toerental regelbaar. Dit laatste is in het simpelste maar energetisch ongunstigste geval mogelijk met een regelbare serieweerstand — een of andere vermogenspotentiometer. Draait de motor echter op minder dan vollast, dan wordt de ongebruikte energie als warmte verstookt in de potmeter. Er zijn nog andere nadelen. Een elektronische 'digitale' aansturing is daarom beter. Puur voor toerenregeling volstaat een halve brug met twee snelle halfgeleiderschakelaars, zoals de typische N-kanaal-MOSFET's in **figuur 7**. De MOSFET's worden aangestuurd met twee PWM-signalen die met elkaar in tegenfase zijn (ze zijn dus elkaars spiegelbeeld) zodat de MOSFET's nooit beide tegelijk geleiden. De gemiddelde spanning is het product van voedingsspanning en de duty cycle. Als bijvoorbeeld de motor in figuur 7 wordt gevoed met 12 V, dan hebben we met het getekende PWM-signaal een duty cycle van 33% (signaal p) en een effectieve voedingsspanning voor de motor van 4 V. De zelfinductie van de motor en zijn mechanische traagheid zorgen ervoor dat bij hogere aanstuurfrequenties de reële blokspanning wordt uitgemiddeld.

U vraagt zich misschien af of hier werkelijk het toerental wordt geregeld en niet simpelweg het vermogen. Maar zo werkt een (asynchrone) elektromotor nu eenmaal: het toerental bij nullast is nooit gewoon een combinatie van toegevoerd vermogen en de daaruit voortvloeiende (mechanische) verliezen. De tegen-EMK oftewel de door de draaiing geïnduceerde tegenspanning is evenredig met het toerental. Die tegenspanning gaat bij nullast dus niet willekeurig omhoog, maar blijft net onder de aangelegde spanning. Het quotiënt van dit kleine spanningsverschil en de effectieve weerstand van de wikkeling(en) levert net genoeg stroom voor een koppel dat de wrijving en andere verliezen compenseert. Wordt de motor dan belast, dan zakt het toerental in, en het verschil tussen aangelegde spanning en tegen-EMK, en dus ook de stroom door de motor, worden groter. De stroom is evenredig met het gewenste koppel; het geleverde vermogen is het product van koppel en toerental.

Uit deze samenhang blijkt ook dat een draaiende motor nooit abrupt kortgesloten mag worden: zijn ohmse weerstand is heel klein, de tegen-EMK is nog groot en de resulterende stroom en het koppel ten gevolge daarvan kunnen onder gegeven omstandigheden tamelijk destructief zijn. Dat geldt ook als we plotseling en abrupt het toerental verlagen met de regelaar in figuur 7. De overtollige kinetische energie moet weg en zorgt voor overbelasting van de transistoren, wat tot defecten kan leiden. En andersom zorgt een snel aanlopen voor hoge aanloopstromen die we in het oog moeten houden.

Volledige brug

Met een volledige brugschakeling (ook wel H-brug genoemd) zoals in **figuur 8** kunnen we zowel toerental als draairichting regelen. Het is in wezen een verdubbeling van de halve brug van figuur 7. Ook de tweede halve brug wordt aangestuurd met PWM-signalen die met elkaar in tegenfase zijn. Er is echter een klein maar wezenlijk verschil: het positieve PWM-signaal p ligt aan de onderste schakelaar T4 en het negatieve signaal ligt aan T3. De motor zit in de middenbalk van de H. Als alle PWM's een duty cycle van 50% hebben, dan is het verschil tussen de gemiddelde spanningen van beide halve bruggen gelijk aan nul en dan staat de motor stil. Met een duty cycle hoger dan 50% op de p-signalen draait de motor de ene kant op. Is dat minder dan 50%, dan draait hij de andere kant op. Een echte verdubbeling met 2 x 2 PWM-signalen voor de aan-

Figuur 9. Met (slechts) drie halve bruggen zijn zogenaamde BLDC-Motoren aan te sturen met de vereiste draaistroom met instelbare amplitude.

Figuur 10. Om (te) hoge aanloopstromen te vermijden worden draaistroommotoren van gemiddeld vermogen eerst opgestart met een sterschakeling en wordt pas bij voldoende hoog toerental omgeschakeld naar driehoekschakeling. Hier ziet u een uitvoering met twee schakelaars.

sturing van de vier transistoren is in principe een luxe. Er bestaan eenvoudiger combinaties, waarbij de linker halve brug met PWM wordt aangestuurd zoals in figuur 7, terwijl de rechterhelft statisch aangestuurd wordt. Anders gezegd: op de linkerhelft wordt het toerental gestuurd tussen 0 en 100% terwijl de aansturing van de rechterhelft de draairichting bepaalt. Maar let op: als we van 100% rechtsom naar 100% linksom (of omgekeerd) willen omschakelen, moeten we eerst de duty cycle voor het p-signaal langzaam van 100% naar 0% regelen, terwijl T4 geleidt en T3 spert. Vervolgens moet T4 sperren en T3 geleiden, en moet de duty cycle van het p-signaal op slag naar 100% gaan, zodat er in eerste instantie 0 volt over de motor staat.

BLDC-aansturing

Ook van meerfasige motoren zijn draairichting en toerental te regelen met PWM. Een klassieker is de borstelloze gelijkstroommotor oftewel BLDC-motor, die in een heleboel hedendaagse aandrijvingen wordt toegepast, van E-bike en Segway tot elektrische auto en nog veel meer. De eenvoudige uitvoeringen hebben drie fasen met echter slechts drie aansluitingen. In tegenstelling tot industriële uitvoeringen van dit type motor, waar meestal alle aansluitingen van de drie wikkelingen naar buiten zijn gevoerd, kan men dan niet kiezen tussen ster- of driehoekschakeling. Maar dankzij de gebruikelijke elektronische aansturing is dat ook niet nodig omdat grote aanloopstromen via PWM kunnen worden opgevangen.

De aansturing van BLDC-motoren is wel iets ingewikkelder dan van enkelfasige motoren, maar dat blijft binnen redelijke grenzen. Vergeleken met figuur 8 ziet u in **figuur 9** maar één extra halve brug. Het venijn zit hem in de aansturingslogica. Ten eerste moeten we voor de draaiing via PWM een draaiveld genereren, en ten tweede is een BLDC-motor in principe een synchoonmotor. Daarom moeten we de spanningen aan de drie wikkelingen commuteren, en wel zodanig dat de drie groepen gespiegelde PWM-signalen voor elke halve brug steeds 120° ten opzichte van elkaar verschoven zijn. Voor een dergelijke elektronische commutatie moeten we dus de verdraaiingshoek van de rotor op elk moment kennen. Dat kan op twee manieren: ofwel de motor is voorzien van (minstens) drie Hall-sensoren, ofwel we benutten de tegen-EMK in de schakelpauzes van de halve bruggen. Met die informatie kunnen we de juiste PWM-signalen aan de drie halve bruggen toevoeren, zodat het veld en daarmee de motor de kant op draait die we willen. Met aansturing via sensoren of EMK hebben we dan uiteindelijk een synchrone aansturing. Met trapeziumvormige wisselspanningen zijn hoge rendementen te behalen terwijl het koppel weinig fluctueert.

Ster en driehoek

Volledigheidshalve: normale draaistroommotoren van middelhoog tot hoog vermogen hebben vaak hoge aanloopstromen. Om die stroom te reduceren kunnen we bijvoorbeeld de motor eerst via een serieweerstand op toeren brengen en dan bij voldoende hoog toerental rechtstreeks aan het net leggen. In de industrie wordt dat inderdaad zo gedaan, met regelbare water-serieweerstanden – de naam geeft al aan waaruit het weerstandsmateriaal bestaat. Bij vermogens tot ongeveer 10 kW volstaat echter het feit dat een motor in sterschakeling maar ongeveer een derde opneemt van het vermogen in driehoekschakeling. Men start de motor dus op in sterschakeling en schakelt om naar driehoekschakeling op het moment dat het toerental meer dan 75% van zijn nominale waarde heeft bereikt.

In **figuur 10** ziet u zo'n schakeling met vermogensrelais. De drievoudige omschakelaar S2 mag onder geen beding van het type 'make before break' zijn. Hij moet eerst onderbreken voordat hij omschakelt, anders gebeuren er ongelukken. Een extra relais met twee omschakelaars kan door het (correct) omwisselen van twee aansluitingen, bijvoorbeeld U1 en U2, de draairichting omkeren.

Zulke ster/driehoekschakelingen zijn in de industrie inmiddels eigenlijk achterhaald, omdat men tegenwoordig motoren met geschikte frequentie-omzetters zacht kan laten aanlopen.

Elektronica

Zoals we al eerder hebben opgemerkt, worden bij moderne motorsturingen voor grote vermogens in de zwareindustrie voornamelijk MOSFET's ingezet als vermogens-halfgeleiders. Rest nog wel de vraag hoe de aansturing met een gespiegeld

Figuur 11. Een willekeurige selectie uit het aanbod van handige controllerprintjes voor motoren met laag vermogen (op eBay).

PWM-signaal concreet moet worden uitgevoerd. In beginsel is het aan te bevelen om de PWM-frequentie zo hoog te kiezen dat die voor mens en dier onhoorbaar is, want anders fungeert de motor deels als luidspreker. Wel neemt wel het schakelverlies toe met toenemende PWM-frequentie. In de regel zien vormen de PWM-frequenties daarom een compromis en liggen tussen 30 en 60 kHz. Verder mogen de twee transistoren van een halve brug nooit of te nimmer gelijktijdig geleiden en daarom moet er tussen de omschakeling van de ene naar de andere altijd een zogenaamde dode tijd liggen. Hoe lang die precies moet zijn hangt af van de eigenschappen van de vermogens-MOSFET's en uiteraard van de PWM-frequentie. Tijden van 0,1...2 µs zijn gebruikelijk. Met een PWM-frequentie van bijvoorbeeld 50 kHz houdt een dode tijd van 0,5 µs in dat de bovenste 2,5% van het maximale vermogen niet gehaald wordt, wat doorgaans aanvaardbaar is. Geschikte gespiegelde PWM-signalen met de juiste dode tijden kunnen, zo men wil, 'met de hand' worden gegenereerd met behulp van een geschikte microcontroller. Met voldoende pootjes op de controller zijn alle zes PWM's voor drie halve bruggen softwarematig te genereren en daarmee hebt u dan alles onder controle. Deze flexibiliteit heeft echter ook nadelen. Als de microcontroller door een bug toevallig mocht vastlopen, dan is niet alleen de kans groot dat de elektronica met enige herrie voorgoed afscheid neemt, maar vooral ook dat er mechanische schade en lichamelijk letsel geleden wordt. Om die reden is het op zijn minst aan te bevelen om speciale halve-brug-drivers voor MOSFET's te gebruiken. Deze dingen bevatten alles voor de aansturing van een typisch N-kanaal-MOSFET-paartje, inclusief het genereren van gespiegelde stuursignalen en (instelbare) dode tijden, evenals leuke details als bootstrapping voor het aansturen van de gates van high-side-FET's (bovenin de schakeling, in figuur 9 zijn dat T1, T3 en T5). Aan de ingang hebben ze slechts een simpel enkelfasig PWM-signaal nodig. Veel driver-chips bevatten alle elektronica voor drie halve bruggen.
De PWM-signalen voor de aansturing van driver-IC's kunnen dan afkomstig zijn van een kleine microcontroller, die dan dus minder te doen heeft. Omwille van de veiligheid moeten we die microcontroller uitsluitend taken laten uitvoeren die puur met de motorsturing van doen hebben, zoals een noodstop-functie, zacht aanlopen, en zo meer. Additionele taken zijn af te raden, ook als er meer dan voldoende rekenkracht beschikbaar is. U kunt ook speciale motor-controller-IC's gebruiken, dan blijft u aan de veilige kant, wellicht ten koste van enige flexibiliteit. Voor kleine motoren tot enige tientallen watt zijn er complete geïntegreerde oplossingen met sturing, drivers en vermogensschakelaars. Dan hoeft u zich alleen nog maar te houden aan de datasheet ;-)
Als u wilt gaan experimenteren met kant-en-klare oplossingen, dan raad ik aan om op eBay de zoekterm 'motor controller' in te geven. Er zijn werkelijk duizenden prints en modules in allesoorten en maten en prijsklassen, van één euro tot meerdere duizenden euro's — voor toepassingen die variëren van eenfase minimotortjes (**figuur 11**) tot industriële aandrijvingen. En bijna alles *made in China*.

180558-03

IN DE STORE

→ **SunFounder PiCar-V voor Raspberry Pi:**
www.elektor.nl/sunfounder-picar-v-for-raspberry-pi

→ **JOY-iT MotoPi – motorsturing voor für Raspberry Pi:**
www.elektor.nl/joy-it-motopi-motor-control-for-raspberry-pi

→ **JOY-iT Motorino – motorsturing voor Arduino:**
www.elektor.nl/joy-it-motorino-motor-control-for-arduino

Weblinks

[1] Stroomoorlog: https://nl.wikipedia.org/wiki/Oorlog_van_de_stromen
[2] Jedlik-motor: http://www.jedliktarsasag.hu
[3] Lohner-Porsche: https://tinyurl.com/y8kx2ldr
[4] Lexikon, figuur links: https://tinyurl.com/y7mn24fz
[5] Lexikon, figuur rechts: https://tinyurl.com/yc2pyp6k
[6] Motor-principe: https://tinyurl.com/yb98ug89
[7] Thyristor: https://tinyurl.com/ychbckro

Instelbare frequentiedeler

met een minimum aan (CMOS-) onderdelen

Michael A. Shustov (Rusland) en **Andrey M. Shustov** (Duitsland)

Er wordt vaak gezegd dat het ontwerpen van een instelbare frequentiedeler een flinke uitdaging is. We zijn deze uitdaging aangegaan en hebben een schakeling ontworpen waarmee de frequentie van een digitaal signaal op een simpele manier door een geheel getal *n* kan worden gedeeld.

De schakeling (zie **figuur 1**) bestaat uit een CMOS XOR-poort type 4070B (IC1.A) en een niet-inverterende buffer type 4050B (IC2.A) die door een instelbaar *RC*-netwerk met elkaar zijn verbonden. Daarnaast is de uitgang van de schakeling teruggekoppeld naar de ingang. De deelfactor *n* is een geheel getal tussen 2 en 1000 (en mogelijk zelfs meer).
De gewenste deling kan met trimpotmeter P1 worden ingesteld. Als de ingestelde waarde van P1 gelijk is aan 0 Ω, dan is *n* gelijk aan 2. Staat de potmeter op zijn maximum waarde (10 kΩ) dan geldt *n* = 94. Verhoging van de waarde van de trimpotmeter naar 75 kΩ geeft *n* = 1000. In het gebied 9 < *n* < 30 is de relatie tussen de ingestelde waarde van P1 en de waarde van *n* vrijwel lineair, waarbij de volgende formule geldt:

$$n = \text{ROUND}\ (P1 + 0{,}2) / 0{,}133$$

Voor P1 wordt de waarde in kΩ ingevuld. Met ROUND wordt een afronding naar het dichtstbijzijnde gehele getal aangegeven. In **figuur 2** ziet u de ingangs- en uitgangssignalen van de frequentiedeler voor de volgende situatie:

- VDD (voedingsspanning voor IC1 en IC2) = 10 V
- Amplitude van het ingangssignaal fin = 10 V
- P1 ingesteld op 2,4 kΩ

Andere deelwaarden kunnen worden verkregen door een aantal van deze frequentiedelers in serie te schakelen (cascadeschakeling). Elke aangesloten frequentiedeler krijgt hierbij zijn eigen poort en buffer in respectievelijk de CD4070BP and CD4050BP IC's. Let op: er moeten gebufferde CMOS-chips worden gebruikt (suffix: B). Ook mag de f_0-ingang niet open blijven, evenals de andere niet gebruikte ingangen van de CMOS-IC's.

180559-03

IN DE STORE
→ Boek: Electronic Circuits For All
www.elektor.nl/electronic-circuits

Figuur 1. Schema van de instelbare frequentiedeler. De (hier niet aangegeven) voedingsspanning V_{DD} van de IC's is 10 V_{DC}.

Figuur 2. De ingangs- en uitgangssignalen van de frequentiedeler bij een deling door *n* = 20.

Zelfbouw-soldeerstation met temperatuurregeling

SMD-solderen voor iedereen

Sunil Malekar (Elektor Labs India)

Het Platino-soldeerstation dat in 2015 werd gepubliceerd, kreeg brede bijval — en niet alleen van het lezerspubliek van Elektor. Alleen al het feit dat het oorspronkelijke project meteen aansloeg, was in de beste ontwerptraditie een belangrijke reden om het oorspronkelijke ontwerp nog eens goed tegen het licht te houden. Alle hieraan bestede inspanningen en ideeën hebben een volledig nieuw ontwerp opgeleverd dat we hier zullen beschrijven.

Het basisidee achter het do-it-yourself SMD-soldeerstation dat in het juli/augustus-nummer 2015 [1] werd gepubliceerd, is niet veranderd: we gebruiken nog steeds de Weller 'RT' soldeerstiften met de handige 3,5 mm jack-connectoren. Deze stiften hebben een ingebouwd verwarmingselement en temperatuursensor. U hoeft dus alleen nog maar voldoende vermogen te leveren om de punt te verwarmen, en te zorgen voor een methode om de temperatuur precies te regelen. Hier is zo'n controller in extreem compacte vorm.

Verwarmen is gemakkelijk

Hoewel we niet precies weten wat er in de soldeerpunt type RT1 zit (**figuur 1**) — helaas maar begrijpelijk maakt Weller deze informatie niet bekend — kunnen we er wel van uit gaan dat het verwarmingselement bestaat uit een resistief element dat warm wordt wanneer er stroom doorheen gaat: hoe meer stroom, hoe heter de punt. Het verhitten van de punt tot een constante temperatuur kan worden gedaan door die stroom te regelen. Omdat de punt een temperatuursensor bevat, is dat vrij gemakkelijk: een gecontroleerde hoeveelheid stroom door de punt sturen en vervolgens de temperatuur meten. Verklein dan de stroom als de temperatuur te hoog is of vergroot die als de temperatuur te laag is.

In het huidige digitale tijdperk waarin nullen en enen de manier bepalen waarop onze constructies (en sommige mensen) werken, wordt een dergelijke regelkring meestal geïmplementeerd met behulp van een microcontroller en een pulsbreedte-gemoduleerd (PWM-)signaal. De gemiddelde waarde van dat PWM-signaal bepaalt de gemiddelde stroom door het verwarmingselement (**figuur 2**).

PWM

PWM-regeling (pulsbreedtemodulatie dus) is om verschillende redenen cool. Allereerst zijn microcontrollers heel goed in het genereren van PWM-signalen — het is een van de dingen die ze het beste doen (hoewel ze soms de kluts kwijt raken op het meest ongelukkige moment), en meerdere kanalen met 16-bits precisie of meer zijn geen enkel probleem. Bovendien zijn PWM-signalen geweldig voor het regelen van (vermogen-)transistoren, omdat ze die snel van *aan* naar *uit* (en omgekeerd) doen schakelen zonder veel tijd te spenderen in het tussenliggende energieverspillende gebied (waar analoge versterkers graag werken). Als gevolg hiervan kunnen vermogenstransistoren grote stromen schakelen zonder heet te worden, en dat is nog meer het geval wanneer het MOSFET's betreft (*Metal Oxide Silicon Field-Effect Transistor*) met hun extreem lage aan-weerstand (R_{DSon}). Samen zorgen deze twee eigenschappen ervoor dat het ontwerp van onze vermogensdriver vooreen soldeerstift relatief eenvoudig blijft.

Meetsignaal en storing gescheiden houden

Het wordt lastiger wanneer we de temperatuur van de tip nauwkeurig willen meten. In tegenstelling tot de aanbevolen ontwerpmethoden voor precisie-meetsystemen, liggen de kabels van de temperatuursensor die het minuscule variërende temperatuursignaal moeten overdragen, zeer dicht bij de draden van het verwarmingselement die een hoogvermogen-signaal voeren. Erger nog, de sensor en het verwarmingselement delen een draad. Hoewel we kunnen begrijpen waarom de soldeerpunt op deze manier is ontworpen, compliceert dit het leven van een systeemontwerper wel een beetje. Het filteren van het uitgangssignaal van de sensor en een zorgvuldige planning van het moment waarop het wordt bemonsterd, verdient daarom aanbeveling!

PROJECT-INFO

solderen
SMD
prototyping

beginners
➡ gevorderden
experts

Ongeveer 4 uur

SMD-soldeergereedschap,
AVR-programmer,
Arduino IDE

Ongeveer € 75

Maximale thermische vermogensregeling

De ontwerpprincipes die we zojuist hebben besproken, zijn onverkort van toepassing op het ontwerp dat hier wordt beschreven,

Figuur 1. De Weller-soldeerpunt type RT1 heeft een spitse tip van 0,2 mm. De 'RT'-serie bestaat uit veel verschillende modellen om precisie-solderen mogelijk te maken.

Figuur 2. Een digitaal gegenereerd PWM-signaal en de corresponderende analoge gemiddelde waarden.

Eigenschappen

- +50 °C tot +450 °C temperatuurbereik
- Nauwkeurige temperatuurregeling
- Verwisselbare RT-series soldeerpunt
- Temperatuur-datalogging
- Externe 12 V_{DC}/2 A voeding

evenals op het vorige op Platino gebaseerde ontwerp. Andere dingen die werden gedeeld met het vorige ontwerp [1] zijn het display — hoewel we dit keer een grafisch OLED-display gebruiken — en de draai-encoder. Nieuw is de meting van de uitgangsstroom, waarmee we de maximale hoeveelheid thermisch vermogen kunnen regelen die de soldeerpunt mag produceren. Dit helpt niet alleen om de tip te beschermen tegen gevaarlijk grote stromen, maar ook om kleine onderdelen te beschermen tegen overmatige hitte.

De schakeling

Na bovenstaande uitleg kan de schakeling — weergegeven in **figuur 3** — gemakkelijk worden begrepen. Laten we voor de verandering onze gedetailleerde beschrijving aan de uitgang beginnen, bij connector K3. Hierop moet de RT1-soldeerpunt worden aangesloten. Pen 1 voert de spanning voor de verwarming, pen 2 het signaal van de temperatuursensor en pen 3 is voor massa.
MOSFET T2 verbindt onder regie van het PWM-signaal periodiek het verwarmingselement in de soldeerpunt met de voedingsspanning voor de verwarming V_{IN} (typisch 12 V). De gate van een MOSFET zoals T2 gedraagt zich bijna als een condensator en daarom is een push/pull-trap, T2/T4, vereist om de poort 'hard' aan te sturen om ervoor te zorgen dat de FET zo snel mogelijk wordt in- en uitgeschakeld, gewoon door op- en ontladen van de gate-capaciteit — zonder dat we ons in allerlei bochten hebben gewrongen om dit te optimaliseren. Transistoren T2 en T4 worden aangestuurd door T3 omdat de microcontroller zelf niet zulke hoge spanningen als V_{IN} (12 V) kan schakelen.
De verwarmingsstroom produceert een kleine spanning over R18, die wordt gekopieerd en versterkt door de schakeling rond IC3, IC2.B en R21, waardoor de ingebouwde analoog/digitaal-omzet-

Figuur 3. Het schema van het soldeerstation combineert op slimme wijze analoge en digitale elektronica om het doel te bereiken.

ONDERDELENLIJST

Weerstanden:
(alle weerstanden SMD 0805, 100 mW, 1% tenzij anders vermeld)
R1,R5,R7,R8,R16,R19 = 10 k
R2,R3 = CG0603MLC-05E, ESD-suppressor, USB-compatibel, SMD 0603
R4,R6 = 22 Ω
R9 = 2k7
R10 = 18 k
R11 = 1 M
R12,R13 = 5k6
R14 = 10 M
R15,R20 = 100 Ω
R17 = 4k7
R18 = 0,02 Ω, 2 W, SMD-stroommeetweerstand
R21 = 100 k
R22 = 560 k

Condensatoren:
(alle condensatoren SMD 0805 tenzij anders vermeld)
C1,C2,C12,C13 = 10 nF
C3 = 10 µF/35 V, case C
C4,C5,C8,C9,C10,C14,C16,C21 ,C23 = 100 nF
C6 = 47 µF/10 V, case A
C7,C18,C20 = 10 µF/10 V, case A
C11 = 22 µF/10 V, case A
C15,C22,C24 = 1 µF
C17,C19 = 22 pF

Spoelen:
L1 = ferriet, 3 A, 30 Ω @ 100 MHz, SMD 0805

Halfgeleiders:
D1 = CD1206-S01575, SMD 1206
D2 = MBRS540T3G
D3 = BAT54
D4 = S1A
T1 = IRF9540NSPBF
T2,T3 = BC847
T4 = BC857
IC1 = NCP1117ST50T3G, SOT-223-3
IC2 = MCP6002-E/SN
IC3 = INA138NA/250G4
IC4 = ATmega32U4-AU

Diversen:
ENC1 = draai-encoder met drukknop, Bourns type PEC11R-4225F-N0024
F1 = PTC, 15 V, I_{hold} = 0,5 A, 4,5 x 3,2 mm
K1 = 2-polige printkroonsteen, steek 5 mm
K2 = micro-USB-connector type B
K3 = 3-polige printkroonsteen, steek 5 mm
K4 = 6-polige (2x3) pinheader, steek 2,5 mm
LCD1 = OLED-display, 0,96", 128x64, SPI-interface
X1 = 16 MHz kristal
6-polige SIL-enkelrijige header, steek 2,5 mm, voor LCD1
6-polige busstrip, 2,5 mm, voor LCD1
Behuizing (suggestie): Teko type 102 (110 x 70 x 46 mm)

Print 180348-1 (Elektor-shop)

ter (ADC) van de microcontroller deze via signaalleiding A1 kan bemonsteren. Condensator C15 zorgt voor enige filtering. Merk op dat de huidige versie van de ATmega-firmware de stroom beperkt tot 1,5 A.
De low-noise versterker IC2.A versterkt de uitgangsspanning van de temperatuursensor van de soldeerpunt. Netwerkjes R12/C12 en R13/C13 vormen samen een tweede-orde laagdoorlaatfilter om een deel van de onvermijdelijke storing "op de lijn" kwijt te raken. Het signaal op lijn A0 wordt bemonsterd door de microcontroller IC4. Weerstanden R11 en R14 maken het mogelijk een niet-aangesloten sensor te herkennen, zodat in dat geval de voedingsspanning naar de soldeerstift kan worden onderbroken. Op deze manier wordt vernieling van de stift door oververhitting voorkomen.

De digitale schakeling

LCD1 is het 0,96-inch OLED-scherm in het project. De bedrading lijkt misschien vreemd, maar vergeet niet dat deze schermen in SPI- of I²C-modus kunnen werken, afhankelijk van hoe sommige weerstanden en jumpers zijn ingesteld. In dit ontwerp wordt het display gebruikt in de SPI-modus. De labels op deze schermen verwijzen echter vaak naar de I²C-signalen; daarom hebben we het getekend zoals in figuur 3. Wanneer u zo'n display voor dit project, zorg er dan voor dat u de SPI-versie krijgt.

ENC1 is de draai-encoder die wordt gebruikt om de temperatuur van het soldeerstation in te stellen. Hij heeft een ingebouwde drukknop en heeft daarom drie digitale ingangen (IO8, IO9 en D6) nodig. IC4 (hiervoor al even genoemd) is de microcontroller die de rekenkacht en het PWM-signaal voor de schakeling levert. Het is een ATmega32U4 met ingebouwde USB-interface. Links van IC4 zien we een micro-USB-connector (K2) die verbinding maakt met deze interface. De onderdelen R2 en R3 maken korte metten met ESD die op deze lijnen kan verschijnen. F1 is een zelfherstellende zekering. Al met al is de USB-poort behoorlijk robuust en het aansluiten van een computer is geen groot risico. Het soldeerstation is ook veilig en zal, wanneer het met de juiste software is geprogrammeerd, als een Arduino Micro of Leonardo op de PC verschijnen.

De voeding

Een 12 V/2 A geregelde voeding is vereist om het soldeerstation correct te laten werken. Hoewel een 12 V DC-mo-

Figuur 4. Bovenzijde van de gemonteerde print met gemonteerde LCD en draai-encoder.

dule goedkoop en handig is, moet u de uitgangsregeling en uitgangstroom controleren voordat u deze gebruikt. Als de voedingsspanning snel inzakt of te veel storing vertoont onder belasting, kunnen de temperatuurmetingen onnauwkeurig worden. IC2 maakt 5 V uit V_{IN} om het digitale gedeelte van schakeling samen met het ADC-ingangscircuit van de temperatuursensor van stroom te voorzien. Componenten L1 en C22 zijn aanvullende maatregelen om storingen buiten de ADC te houden.

Voor ontwikkelingsdoeleinden is het ook mogelijk om het circuit vanaf de USB-poort van stroom te voorzien. Wanneer V_{IN} niet is aangesloten, verbindt D3 VUSB met de 5V-lijn, waardoor de schakeling effectief wordt gevoed. De soldeerpunt zal in dit geval niet warm worden, maar temperatuurmetingen zijn mogelijk — en dat betekent dat u deze schakeling ook als een thermometer kunt gebruiken!

Figuur 5. Componentenzijde van de opgebouwde print inclusief alle SMD-onderdelen.

Enkele software-details

Het programma dat alles mogelijk maakt, neemt verschillende zaken voor zijn rekening:

- genereren van het PWM-signaal;
- temperatuurmeting;
- meting van de verwarmingsstroom;
- meting van de spanning over het verwarmingselement;
- USB-communicatie;
- interactie met de gebruiker.

Na het opstarten zorgt het programma ervoor dat de tip de volle voedingsspanning krijgt om een snelle start te garanderen. Zodra de doeltemperatuur is bereikt — dit duurt een paar seconden — begint de PWM-regeling en kunt u beginnen met solderen. Merk op dat de doeltemperatuur wordt opgeslagen in het EEPROM van de MCU om te voorkomen dat deze telkens opnieuw moet worden ingesteld als het soldeerstation wordt ingeschakeld.

De temperatuur van de soldeerpunt wordt periodiek gemeten. Om dit goed te doen, wordt het PWM-signaal eerst uitgeschakeld (laag gemaakt) en na een korte vertraging (om de wijziging effect te laten hebben), wordt het signaal op A0 bemonsterd. De momentele temperatuur wordt berekend en vergeleken met de ingestelde waarde, en als het verschil te groot is (ongeveer 5 °C), wordt de PWM-waarde dienovereenkomstig aangepast. In tegenstelling tot de temperatuurmeting wordt de meting van de verwarmingsstroom uitgevoerd wanneer het PWM-signaal hoog is (dat wil zeggen aan) omdat anders de waarde altijd nul of daaromtrent zou zijn.

De gebruikersinterface (UI) bestaat uit de draai-encoder en het display. Voor de eerste pin-change (PC) worden interrupts gebruikt; het proces wordt beheerd met behulp van de populaire Adafruit grafische bibliotheek.

Zowel ingestelde als gemeten temperatuur alsmede de status van het volledige soldeerstation worden verzonden via de seriële USB-poort.

Het programmeren van de firmware

De software voor IC4 (zie [2]) is geschreven als Arduino-sketch. Daarom hebt u voor het compileren de Arduino IDE nodig. Omdat het soldeerstation Arduino Micro (Leonardo)-compatibel is, is dat ook het "board" dat in de IDE moet worden geselecteerd.

K4 is bedoeld voor het programmeren van de microcontroller. Zodra u de Arduino Micro- of Leonardo-bootloader hebt geflashed, kunt u de MCU ook via USB vanuit de Arduino IDE programmeren.

Figuur 6. Close-up van de 0,02- ohm, 2-watt stroommeetweerstand op de print van een vroeg prototype. Afgebeeld is een geïmproviseerd exemplaar met een through-hole 0,02-ohm-vervanger als vervanging voor een echt "AORS1"-SMD-exemplaar van Welwyn dat niet op voorraad was toen de foto werd gemaakt. SMD of niet, het werkt.

De opbouw

U kunt ervoor kiezen om een kit met voorgemonteerde SMD's te kopen in de Elektor-shop, of de onderdelen bij uw favoriete leverancier(s) en de print bij Elektor te bestellen en een middagje aan het solderen van SMD's te besteden.

Het assembleren van het bord vereist SMD-gereedschappen; de foto's van de **figuren 4 en 5** van de compleet gemonteerde print kunnen nuttig zijn om het project thuis na te bouwen. Een speciaal onderdeel is de 0,02-Ω-stroommeetweerstand van **figuur 6**, die aan de zijkant van de print is gemonteerd bij de driepolige printkroonsteen. Die weerstand is niet meer dan een metalen strip die zo is gebogen dat hij met 'pootjes' iets boven de print kan worden gemonteerd. De lengte van soldeereiland tot soldeereiland legt de waarde behoorlijk nauwkeurig vast.

De resulterende 'module' is vrij klein en past gemakkelijk in allerlei behuizingen, waarbij voldoende ruimte overblijft voor een 12 V/2 A voedingsmodule of een transformator plus gelijkrichter, stabilisator enzovoort zodat het geheel niet te licht is. Het prototype van ons soldeerstation is ingebouwd in een Teko-behuizing type 102 met afmetingen 110 x 70 x 46 mm.

Om diezelfde reden is het misschien geen gek idee de print voor het soldeerstation in een zware metalen behuizing in bouwen. Als de behuizing te licht is, verschuift deze gemakkelijk, terwijl u eigenlijk wilt dat deze op uw werkblad blijft staan. Een goede oplossing zou daarom zijn om de module met zijn voeding — rond een zware transformator — in één behuizing in te bouwen.

Wanneer u kiest voor een afzonderlijke 12 V/2 A-voeding voor het soldeerstation, moet u alle elektrische veiligheidsmaatregelen naleven die voor u van toepassing zijn. Daarom beveelt Elektor Labs het gebruik aan van een ingegoten 12-V DC-voeding van het type dat wordt gebruikt voor printers en lampen, omdat daarmee gevaarlijke situaties grotendeels worden voorkomen. In de plaatselijke kringloopwinkel kunt u ongetwijfels een geschikt exemplaar vinden!

Een andere (kleinere) uitdaging is om een kabel te vinden die:

- flexibel is;
- hittebestendig is;
- gespecificeerd is voor de nominale stroom;
- en die aan 3,5-mm-jackpluggen gesoldeerd kan worden.

Wij vonden USB-kabels heel bruikbaar — snij de connectoren eraf en gebruik twee aders voor massa en voeding.

Een idee is om een connector op de behuizing te monteren en vandaar draden naar de module te leiden. De soldeerpunt wordt aangesloten op de kabel en de kabel wordt op zijn beurt weer aangesloten op die aansluiting.

Kortom, bouw uw doe-het-zelf soldeerstation op als een commercieel exemplaar; er zijn goede redenen waarom die allemaal dezelfde opzet hebben.

Gebruik van het soldeerstation

Het soldeerstation is lekker gemakkelijk in het gebruik, het enige wat u moet doen, is voor een voeding zorgen en met de encoder een gewenste temperatuur instellen. Elke waarde tussen 50 °C en 450 °C kan worden ingesteld. De ingestelde temperatuur wordt samen met de gemeten (werkelijke) temperatuur van de soldeerpunt op het display weergegeven.

Als de encoder ongeveer 15 minuten niet is gebruikt, zal het soldeerstation overgaan in de energiesparende modus. Wanneer u de encoderknop ongeveer 10 seconden indrukt, gaat het soldeerstation eveneens in de slaapstand. ⏮

180348-04

IN DE STORE

→ **DIY soldeerstation met regelbare temperatuur, kale print:**
www.elektor.nl/diy-solder-station-bare-pcb

→ **DIY soldeerstation met regelbare temperatuur, kit met alle onderdelen:**
www.elektor.nl/diy-solder-station-kit-of-parts

→ **DIY soldeerstation met regelbare temperatuur, geprogrammeerde microcontroller:**
www.elektor.nl/diy-solder-station-controller

Weblinks

[1] Platino-based Solder Station, oorspronkelijke publicatie: www.elektormagazine.nl/140107

[2] Downloads bij dit artikel: www.elektormagazine.nl/180348-04

electronica
een stukje

Clemens Valens (ElektorLabs)

De tweede editie van electronica Fast Forward, the startup platform powered by Elektor — georganiseerd en gepresenteerd in samenwerking met 's werelds belangrijkste elektronicabeurs in München — zullen we ons blijven herinneren door het hoge technische en professionele niveau van de deelnemende bedrijven en hun projecten.

32 deelnemers uit niet minder dan 10 landen kregen ieder vijf minuten om hun project te pitchen voor een jury die bestond uit experts van (in alfabetische volgorde) Arrow, Avnet-Silica, EBV Elektronik, en Elektor. Na drie enerverende presentatiedagen bleven zes finalisten over: Teiimo, Querom, V-Juice, ValCUN, Volabo, en Wizama. Een dag later werd de 'wildcard' toegekend aan Wisebatt, wat het aantal finalisten op zeven bracht. Na de finaleronde werd Teiimo — een start-up gespecialiseerd in wearable elektronica en 'smart'-textiel — unaniem tot winnaar uitgeroepen.

Het driedaagse evenement, dat werd gesponsord door Arrow, Avnet-Silica & EBV Elektronik (diamant-sponsors) en Trinamic (brons-sponsor), bood zowel het publiek als de jury een indrukwekkende en interessante blik op wat de toekomst ons kan brengen. De deelnemers kwamen niet alleen uit Europa maar ook zelfs uit Australië en de Verenigde Staten, en de onderwerpen varieerden van het 3D-printen van metalen objecten, interactieve bordspellen

Markus Strecker van **Teiimo GmbH** — Technologies in Motion, winnaar van electronica Fast Forward, the startup platform powered by Elektor 2018 — ontving de Eerste Prijs uit handen van electronica CEO Dr Martin Lechner.

Met hun innovatieve interactieve bordspel hoopt de winnaar van de tweede prijs, **Wizama**, de gezellige familiesfeer te laten herleven die onze ouders en grootouders zich zullen herinneren uit de tijd vóór de smartphones en tablets.

Fast-Forward 2018

toekomstvisie van ‘smart’-start-ups

De winnaars

1. € 75,000 media-budget en een eigen stand op electronica 2020: **Teiimo GmbH (Duitsland) — Wearable Electronics and Smart Textiles**
2. € 50,000 media-budget: **Wizama (Frankrijk) — Interactive Board Game Console**
3. € 25,000 media-budget: **Querom Elektronik GmbH (Duitsland) — High-voltage DC/DC Converter Modules**

Eervolle vermelding

€ 750: Max & Michael Buschmann — **Cocktail Joe Automatic Cocktail Machine** *(Wij willen er ook een!)*

en laser-geprinte rekstrookjes tot snelle online-prototyping-diensten, cocktail-robots en virtual-reality-systemen. Het was ons een eer om deze geïnspireerde mensen te ontmoeten en te horen wat hun plannen zijn voor de ontwikkeling van elektronische producten. Bent u er de volgende keer ook bij?

180702-03

Web Link

[1] electronica Fast Forward 2018 webpagina: www.elektormagazine.com/labs/contest/e-ffwd-contest

De winnaar van de Derde Prijs, **Querom Elektronik GmbH**, is gespecialiseerd in op maat gemaakte vermogenselektronica en hoogspannings-DC/DC-converters met uitgangsvermogens van 1 tot 10 kW.

Cocktail Joe, een geautomatiseerde cocktailmixer die op feestjes absoluut de aandacht zal trekken.

de Elektor-community

LEARN DESIGN SHARE

83	248342	1045	514	250000
landen	leden	experts & auteurs	publikaties	maandelijkse bezoekers

Elektor is meer dan een magazine: het is een community van actieve elektronica-ingenieurs (van beginners tot pro's) die allemaal willen leren, ontwerpen en delen met vakgenoten.

Elektor Web Store: de 24/7 snoepwinkel voor iedere electronica-ingenieur!
10% korting voor Green- en Gold-members:
www.elektor.nl

Elektor Magazine: zes maal per jaar een lekker dik blad in de bus
Boordevol electronica-nieuws, reviews, tips- en trucs: **www.elektormagazine.nl**

Elektor PCB Service: bestel uw eigen printen!
Zowel voor prototypes of grotere aantallen:
www.elektorpcbservice.com

Elektor wekelijks & digitaal:
uw gratis wekelijkse e-zine.
www.elektor.nl/inschrijven

Elektor Academy: webinars, seminars, presentaties, workshops, DVD's...
Praktijkgericht leren:
www.elektor.nl

Elektor Boeken: o.a. Arduino, Raspberry Pi, ESP32, IoT en Linux
Verkrijgbaar in onze Web Store met 10% leden-korting: **www.elektor.nl/boeken**

Elektor TV: reviews, time lapse, unboxing en eigen Journaals
Kijken is leren:
www.elektor.tv

Elektor Labs: eigen projecten tonen, leren van anderen
Wij ontwikkelen en testen zelf!
www.elektormagazine.nl/labs

Plug in op onze community

GREEN € 1,78/week

- ✔ Toegang tot ons web-archief
- ✔ 10% korting in onze webshop
- ✘ 6x Elektor Magazine (Print)
- ✔ 6x Elektor Magazine (PDF)
- ✔ Exclusieve aanbiedingen
- ✔ Toegang tot meer dan 1000 Gerberfiles
- ✘ Elektor's jaarlijkse DVD-ROM

WORD LID

www.elektor.nl/green-lidmaatschap

GOLD € 2,45/week

AANRADER

- ✔ Toegang tot ons web-archief
- ✔ 10% korting in onze webshop
- ✔ 6x Elektor Magazine (Print)
- ✔ 6x Elektor Magazine (PDF)
- ✔ Exclusieve aanbiedingen
- ✔ Toegang tot meer dan 1000 Gerberfiles
- ✔ Elektor's jaarlijkse DVD-ROM

WORD LID

www.elektor.nl/gold-lidmaatschap

GRATIS

- ✘ Toegang tot ons web-archief
- ✘ 10% korting in onze webshop
- ✘ 6x Elektor Magazine (Print)
- ✘ 6x Elektor Magazine (PDF)
- ✔ Exclusieve aanbiedingen
- ✘ Toegang tot meer dan 1000 Gerberfiles
- ✘ Elektor's jaarlijkse DVD-ROM

WORD LID

www.elektor.nl/inschrijven

facebook.com/ElektorNL twitter.com/ElektorNL instagram.com/elektorlabs

Obsolete of ‘allocated’?

Jan Visser (Elektor Labs)

Woensdagmiddag rond 3 uur: nog even wat componenten bestellen voor enkele lopende projecten. Meteen maar even de voorraad nalopen van onze SMD-componenten. Na controle blijkt gelukkig dat we alleen wat standaardcomponenten missen in onze SMD-bakjes. Wat 0805-weerstandjes van 1 k, 10 k, 100 k enzovoort, en dan nog wat standaard-condensatortjes 0805 in de waarden 1 nF, 10 nF, 100 nF. Voor de projecten heb ik enkele ‘zware’ controllers en andere exoten op mijn verlanglijstje staan, waarvan ik vermoed dat ik wel wat leverproblemen zal tegenkomen, maar goed: eerst maar eens de standaardcomponenten inkloppen bij de grote bekende leveranciers, dan heb ik dat alvast gehad.

Twee uur en enkele koppen koffie later blijkt dat niet de exoten en controllers het probleem vormen, maar dat juist de ‘ordinaire’ componenten moeilijk verkrijgbaar zijn. Waar de exoten en ‘moeilijke’ controllers juist in allerlei packages en in kleine en grote aantallen leverbaar zijn bij diverse leveranciers, is het met de verkrijgbaarheid van ‘standaard’-componenten treurig gesteld!

Normale 0805-SMD-weerstandjes en -condensatoren hebben opeens levertijden van enkele weken tot enkele maanden, of de levertijd wordt aangegeven in termen van ‘wordt verwacht’! Zonder datum er bij, dus het kan verkeren... Sommige leveranciers sturen je keurig een bericht als je een component besteld hebt, met een te verwachten levertijd. Zo kreeg ik laatst nog een email over een 1-µF-condensatortje dat binnenkort weer leverbaar was: verwachte leverdatum eind 2019! Navraag bij verschillende leveranciers en even rondbellen met contacten in de industrie leert dat de voorraden normale componenten massaal opgekocht worden door producenten van mobiele telefoons, televisies en andere elektronica, om aan de groeiende behoefte van de consument te voldoen. En dan heb je nog de zogenaamde brokers, de tussenhandelaren, lieden die schaarse voorraden opkopen en daarmee gaan handelen als waren het beursaandelen. Een nadeel van de economische groei, zullen we maar zeggen. Maar juist deze groep brokers, die voorraden opkopen, in een magazijn bewaren totdat de vraagprijs ervoor is verdubbeld of zelfs vertienvoudigd, heeft niets met elektronica. Zij willen alleen maar zoveel mogelijk met hun investering verdienen. Helaas zorgen zij met hun opkoopbeleid er wel voor dat zelfs de meest gangbare componenten ‘obsolete’ (niet meer gangbaar) lijken te zijn. In werkelijkheid zijn deze componenten er wel degelijk, maar niet op de juiste locatie: ‘allocated’ dus. Dit alles maakt het voor de redelijk gewone stervelingen onder ons, of een klein lab of makerspace, moeilijk om aan onderdelen te komen, tenzij je gaat onderhandelen over tienduizenden stuks of bereid bent exorbitante prijzen te betalen voor enkele tientallen weerstandjes.

Ik ben al heel lang bij Elektor werkzaam, en moest in dit verband denken aan een artikel uit januari 2006, waarin gewaarschuwd werd voor schaarste op de componentenmarkt en aangeraden werd om IC’s, controllers en exoten van gebruikte printen af te solderen om ze daarna te hergebruiken. In dat artikel werd gesteld dat het demonteren van (bijna) waardeloze componenten zoals weerstanden en condensatoren niet interessant was, omdat die altijd probleemloos verkrijgbaar zijn.

Hoe anders kijken we nu tegen deze situatie aan: in sommige grote steden van China en India zijn er al complete stegen waarin talloze mensen de hele dag bezig zijn met het desolderen van allerlei SMD-componenten om die vervolgens weer als ‘refurbished’ te verkopen in de winkel aan de voorkant van de straat. Ook de eenvoudige standaardweerstandjes en condensatortjes worden daarbij niet vergeten. Misschien zien we binnenkort dit fenomeen ook opduiken in de straten van Amsterdam, London of München? Het loont dus ook in deze tijd om je oude projecten niet weg te gooien, maar af en toe tevoorschijn te halen om er hier en daar een simpele component af te solderen. Waarbij natuurlijk wel de vraag rijst: heb ik die component al ergens gebruikt voor een ander project? En zo ja, waar is dat printje dan gebleven? En daarmee is de cirkel rond, want de vraag blijft hetzelfde: obsolete of allocated?

Vriendelijke groeten uit het Elektorlab
Jan Visser

180308-01

HOMELAB PROJECT

Vuurtorenlicht 2.0

Met instelbaar lichtkarakter

Friedrich Lischeck (D)

In het novembernummer van 2010 presenteerde Elektor een schakeling van Leo Szumylowycz, dat het roterende baken van een vuurtoren simuleerde [1]. De schakeling gebruikte enkel analoge componenten, zoals de legendarische 555-timerchip en een standaard LM358-opamp. Het werkte perfect. Het enige nadeel was het aantal benodigde componenten — in totaal 29. Met een ATtiny45-microcontroller kunnen de uitgaven aan onderdelen drastisch worden verlaagd. De hier gepresenteerde schakeling heeft slechts 8 componenten nodig voor een uitgebreide reeks functies.

Het ronddraaiende licht van een vuurtoren neem je waar als een langzame toename van de lichtsterkte wanneer de bundel naar je toe beweegt. Dan volgt er een korte, felle flits als hij recht op jouw locatie staat gericht, en vervolgens neemt de lichtsterkte weer af. En dit alles in een voor de vuurtoren kenmerkend *lichtkarakter* (dus met kortere en langere tussenpozen). Elektrisch is dit te benaderen door een driehoeksignaal met op het maximum een sterke impuls gesuperponeerd. In dit project realiseren we dit via een in BASCOM geprogrammeerde microcontroller.

Figuur 1. Het schema is met slechts 8 onderdelen uiterst eenvoudig te noemen.

Hardware

De schakeling is zeer eenvoudig, zoals zo vaak het geval is met microcontrollers: 2 weerstanden, 1 condensator, 3 potentiometers, 1 led en de microcontroller. Dat is alles wat je nodig hebt (zie **figuur 1**). Het meeste werk zit in de software. We kunnen hier dus heel beknopt zijn. De drie potentiometers zijn aangesloten op de analoge ingangen ADC1, ADC2 en ADC3 van de ATTiny45. Weerstand R1 zorgt voor een schone reset. R2 beperkt de stroom door de led (D1) en C1 onderdrukt eventuele storingen.

De volgende parameters kunnen met de drie potentiometers worden ingesteld:

- P1: De virtuele vuurtoren kan knipperen (licht aan en uit) of flitsen (tot een bepaalde intensiteit opzwellen, dan plotseling flitsen en dan weer in intensiteit afnemen, zoals de schakeling van Leo Szumylowycz ook deed). P1 bepaalt de verhouding tussen het aanzwellen en uitdoven en de heldere flits, waarmee je dus ook de duur van de flits instelt. In de twee eindposities krijg je ofwel alleen maar zwelling en verval of alleen knipperen. De totale duur van een cyclus is altijd één seconde.

- P2: De volledige cyclus van oplichten en uitdoven wordt het *lichtkarakter* genoemd. Als een licht bijvoorbeeld 1 seconde aan, 1 seconde uit, 1 seconde aan en 10 seconden uit is en dan opnieuw begint, spreek je over een lichtkarakter van 13 seconden. In tegenstelling tot de oorspronkelijke schakeling uit 2010, is het met P2 mogelijk om de duur van het lichtkarakter in te stellen.

- P3: Met P3 stel je het aantal flitsen in het lichtkarakter in. Ook deze functie bestond niet in de oorspronkelijke schakeling. Als het lichtkarakter korter is dan de totale tijd die nodig zou zijn voor de ingestelde flitsen, wordt de duur van het lichtkarakter automatisch ingesteld op de totale duur van de ingestelde flitsen + 2 seconden.

Met deze instelmogelijkheden kun je veel bestaande vuurtorens simuleren.

Software

Zoals gezegd, zit het leeuwendeel van het werk in de software. Het programma is geschreven met BASCOM-AVR. De broncode is beschikbaar via [2]. We lichten hier een aantal van de meest interessante programmaonderdelen toe.
Voor het Vuurtorenlicht 2.0 zijn zowel Timer0 als Timer1 vereist. Timer0 regelt de pulsbreedtemodulatie waarmee de helderheid van de led wordt gestuurd. Timer1 is de klokgenerator waarmee de gewenste oplichttijden worden gerealiseerd. Het programma bevat een bijzonderheid in de timerprogrammering. Omdat BASCOM Timer1 van de ATtiny45 niet aan kan sturen, wordt hier direct register TCCR1 van de ATTiny45 aangesproken:

```
Preload1 = 99 'komt overeen met 0,01 seconden
TCCR1 = &H0A 'staat voor Config Timer1=TIMER,
   Prescale = 512
on Timer1 Zeitbasis1
enable Timer1
enable Interrupts
```

Opvallend is dat een prescale-waarde van 512 mogelijk is, een waarde die niet mogelijk is bij andere hedendaagse microcontrollers van Atmel zoals de ATMega8. `Zeitbasis1` ('tijdbasis') is een interruptgestuurde tijdroutine. Timer0 wordt zoals gezegd gebruikt voor pulsbreedtemodulatie en wordt op de normale wijze ingesteld door BASCOM:

```
Config Timer0 = Pwm , Compare_A_Pwm = Clear_Up ,
   Compare_B_Pwm = Clear_Up , Prescale = 1
Enable Timer0
```

`Zeitbasis1` is de kern van het programma. Laten we er eens een kijkje in nemen (zie **Listing 1**). De interruptroutine `Zeitbasis1` gebruikt de variabelen `LEDan` ('led aan'), `Blitzstart` ('start flits') en `Blitzende` ('stop flits'), die in het hoofdprogramma gegenereerd worden. Ook gebruikt de routine het variabelenveld `Form(N)`, dat eenmalig gegenereerd wordt in het hoofdprogramma. Het geeft de variabele `Zehntel` ('tiende') door aan het hoofdprogramma.
De interruptroutine wordt elke honderdste van een seconde opgeroepen en is verdeeld in twee delen. In het eerste deel worden de variabelen `Hunderstel`, `N` en `Zehntel` telkens verhoogd. `Hundertstel` staat voor een honderdste seconde en

Listing 1

```
Zeitbasis1:
 Timer1 = Preload1
  incr Hundertstel
  incr N
  if Hundertstel>9 then
   Hundertstel=0
   Incr Zehntel
 end if

 if N >100 then N=1
 if LEDAn=1 then
  Ausgabe=Form(n)
  if n>= Blitzstart and n<= Blitzende then
   Ausgabe=255
  OCR0A= Ausgabe 'Nieuwe helderheid instellen
 else
  OCR0A=0
 end if
Return
```

`Zehntel` staat voor een tiende seconde. `N` staat ook voor honderdste seconden, maar wijkt in die zin af dat `N` een waarde van 1 tot 100 kan aannemen, terwijl `Hundertstel` van 0 tot 9 gaat en slechts dient om de tienden van seconden op te laten lopen in de variabele `Zehntel`.
In het tweede deel van de routine wordt de helderheidsregeling van de led afgehandeld. Elke 100ste seconde, vertegenwoordigd door variabele `N`, wordt een nieuwe helderheidswaarde ingesteld als de led op dat moment aan is, wat wordt aangegeven door de variabele `LEDAn` van het hoofdprogramma. De helderheidswaarden zijn opgeslagen in `Form(N)`, waarbij `N` elke 100ste seconde met één toeneemt. Alle 100 waarden van `N` bij elkaar resulteren dus in 1 seconde — de duur van de ingeschakelde toestand van de led. De helderheidswaarde in `Form(N)` wordt telkens overgenomen in de variabele `Ausgabe` ('uitgang').
Met het bovenstaande neemt de helderheid van de led alleen toe en af. Nu voegen we variabelen `Blitzstart` en `Blitzende` toe. Deze bevatten ook waarden tussen 1 en 100, in overeenstemming met 100ste seconden tijdens de ingeschakelde toestand van de led. Als `N` groter dan of gelijk aan de waarde van `Blitzstart` en kleiner dan of gelijk aan de waarde van `Blitzende` is, licht de led op met de maximale waarde ('255'). Deze waarde overschrijft de op dat moment ingestelde waarde van de variabele `Ausgabe`. Met `OCR0A= Ausgabe` wordt de nieuwe helderheidswaarde (uitgedrukt in duty-cycle) voor pulsbreedtemodulatie ingesteld.
Zoals reeds vermeld, gebeurt dit alleen wanneer `LEDAn` de ingeschakelde toestand aangeeft. Wanneer `LEDAn` de uitgeschakelde toestand aangeeft, gebeurt dit alles niet en wordt de waarde van de pulsbreedtemodulatie met `OCR0A=0A=0` op nul ingesteld.

Hoofdprogramma

Het hoofdprogramma loopt in een eindeloze DO-LOOP en bevat twee hoofdfunctiegroepen. In het eerste deel worden de waarden van de drie potentiometers P1-P3 uitgelezen, aan de drie

Listing 2

```
If Zehntel <> Lzehntel Then
 Incr X
  If X = Wiederkehr Then
   X = 1
   disable Interrupts
   N = 0
   enable Interrupts
  end if
  if X < Kennungsdauer then
   If X = Naechste Then
    Naechste = Naechste + 10
    if LEDAn = 1 then
     LEDAn = 0
    else
     LEDAn = 1
    end If
   end If
  End if
  If X = Kennungsdauer then
   LEDAn= 0
   Naechste=1
  end if
 Lzehntel = Zehntel
End If
```

bijbehorende variabelen `Blitzstart`/`Blitzende`, lichtkarakter en identificatieduur toegewezen en genormaliseerd naar de benodigde waarden. Dit gebeurt echter alleen als de variabele `LEDAn` de waarde nul bevat.
Het tweede deel van het hoofdprogramma (zie **Listing 2**) regelt het in- en uitschakelen van de `LEDAn`-variabele met de gewenste timing. Want er moet niet bij *iedere* do-lus een actie volgen. De bedoeling is dat er slechts één maal per tiende seconde een actie volgt. Dit wordt bewerkstelligd door de twee variabelen `Zehntel` en `LZehntel`. De naam `LZehntel` staat voor de laatste tiende waarbij een actie plaatsvond.
`LEDAn` wordt per interval van één seconde op één of nul ingesteld. Het programma maakt hiervoor gebruik van de variabelen `Naechste` ('volgende') en `X`. `X` wordt elke tiende van een seconde met 1 verhoogd en pas bij het bereiken van de waarde het lichtkarakter ('`Wiederkehr`') naar 1 gereset. Daarbij wordt de variabele `N` van interruptroutine `Zeitbasis1` gesynchroniseerd en op 0 gezet. Om ervoor te zorgen dat dit niet gebeurt tijdens een actie van de interruptroutine, worden interrupts van tevoren gedeactiveerd en daarna weer vrijgegeven.
Als `X` de waarde van `Naechste` heeft bereikt, d.w.z. als er een seconde is verstreken, wordt de variabele `LEDAn` omgezet. Bovendien wordt de waarde van `Naechste` met 10 verhoogd, wat overeenkomt met het instellen van de volgende actie over één seconde. Dit gebeurt echter alleen zolang `X` kleiner is dan de waarde van de identificatieduur ('`Kennungsdauer`'). Als deze wordt overschreden, wordt `LEDAn` op 0 gezet en `Naechste` op waarde 1, zodat `LEDAn` pas aan het begin van een nieuwe cyclus wordt gewijzigd.

Voor het verloop van de ingeschakelde toestand wordt gebruik gemaakt van volgende code:

```
for I=1 to 10: Form(I)=1:Next I
for I=11 to 20: Form(I)=2:Next I
for I=21 to 30: Form(I)=3:Next I
for I=31 to 40: Form(I)=4:Next I
for I=41 to 50: Form(I)=5:Next I

For I = 51 To 100
 J = 101 - I
 Form(i) = Form(j)
Next I
```

De waarden voor de elementen van 1 tot 50 worden in stappen van 10 ingesteld. De waarden van de elementen 51 tot 100 worden achterstevoren overgenomen van deze waarden (dus

Figuur 2. Hier zijn de fuse-instellingen voor de ATtiny45 te zien zoals ze in AVR-Studio 4.18 worden weergegeven.

Figuur 3. De schakeling kan eenvoudig op een breadboard opgebouwd worden.

van 50 naar 1). Uiteindelijk benadert de variabele `Form(I)` dus een driehoekscurve met een maximum van 5.

Om de schakeling goed te laten werken, moeten de fuses voor de ATtiny45 op de volgende waarden worden ingesteld:

```
EXTENDED: 0xFF
HIGH : 0xDF
LOW : 0xE2
```

Zie ook **figuur 2**.

Opbouw

De montage kan op een gaatjesprint gedaan worden of, zoals in **figuur 3** te zien is, op een breadboard. Het is raadzaam om een IC-voetje te gebruiken. Zeer praktisch zijn de voetjes met geïntegreerde condensator, dan hoeft C1 — net als in figuur 2 — niet geplaatst te worden.

Instellen

Stel eerst alle drie de potentiometers in op de laagste waarde. Stel nu met P1 de gewenste flitslengte in. Vervolgens is het aan te bevelen om het aantal flitsen per cyclus in te stellen met P3. Omdat het lichtkarakter nog op de minimale tijdsduur staat, maakt het programma dit automatisch 2 seconden langer dan de flitsen en het oplichten samen met de bijbehorende onderbrekingen vereisen. Nadat je het aantal flitsen en het oplichten hebt bepaald, gebruik je P2 om de duur van het lichtkarakter in te stellen op de gewenste waarde. Nu zijn de parameters van de vuurtoren ingesteld.

(180334)

IN DE STORE

→ E-Book: Microcontroller Basics
www.elektor.com/microcontroller-basics-ebook-en

→ Boek: Make: AVR Programming
www.elektor.com/make-avr-programming

Weblinks

[1] Elektor november 2010, www.elektormagazine.nl/magazine/elektor-201011/16200

[2] Downloads: www.elektormagazine.nl/180334-01

Advertentie

Steun ons en ontvang een cadeau dat u hopelijk nooit nodig hebt!

Wanneer u donateur wordt en ons steunt voor slechts €7 per maand, krijgt u naast onze hartelijke dank ook een gratis brandblusser.
Zolang u donateur blijft, zorgen wij ervoor dat deze gegarandeerd werkt en om de drie jaar vervangen wordt.

Met uw steun realiseren we onze drie voornaamste ambities:

1. **Een brandveilig Nederland waar je met een minimumrisico op brand(wonden)letsel kunt wonen, werken en leven.**
2. **Uitstekende behandeling van patiënten met brandwonden gericht op een snel en zo volledig mogelijk herstel.**
3. **De allerbeste nazorg, gericht op re-integratie van de patiënt tot de hoogst mogelijke maatschappelijke participatie.**

nederlandse brandwonden stichting

Steun ons via
www.elektormagazine.nl/brandwondenstichting

ESP32-Weerstation

Sensordata online uitlezen

Roy Aarts (Elektor Labs) & **Thijs Beckers** (Redactie)

Het ESP32-board is een veelzijdig en zeer betaal ontwikkelbordje dat zich heel goed leent voor allerlei domoticaprojecten. Hier presenteren we een toegespitste toepassing waarin de ESP32 het hart vormt van een weerstation dat alle data direct online beschikbaar maakt.

Ons prototype past perfect in de gekozen behuizing. Let op: hier is nog een standaard 7805 op de print gezet in plaats van de DC/DC-converter

Eigenschappen

- meet temperatuur, windrichting en –snelheid, luchtvochtigheid, luchtdruk en neerslag
- optionele sensor voor fijnstof: Nova Fitness SDS-011
- extra aansluitingen voor bijvoorbeeld *Grove*-sensoren
- werkt met Thingspeak en senseBox
- in te stellen via interne webpagina van de ESP32
- werkt op zonnepaneel, 12V-accu en/of 8-28V-DC-adapter

Het ESP32-Weerstation meet de meest gangbare weersomstandigheden: temperatuur, windrichting en –snelheid, luchtvochtigheid, luchtdruk en neerslag. Met de juiste sensor, bijvoorbeeld de SDS-011 van Nova Fitness [1] (PM2.5, resolutie 0,3 µg/m^3), meet hij ook de concentratie fijnstof. De temperatuur, luchtvochtigheid en luchtdruk meet hij met

een BME280-sensor van Bosch [2]. Voor de winsnelheid, -richting en neerslag hebben we een weerstation-set ingezet die ook in onze shop te vinden is (zie het kader 'In de Store').
De ESP32 voert de metingen uit en kan ze daarna uploaden naar Thingspeak [3] of senseBox [4]. Thingspeak is een online database van Mathworks, waar de gegevens naartoe geüpload en netjes als grafiek afgebeeld kunnen worden. De gegevens kunnen ook met Matlab worden bewerkt.
SenseBox functioneert als een soort open source weeroverzichtskaart. Gebruikers die een senseBox-kit of ander senseBox-compatibel apparaat hebben, kunnen hun metingen uploaden, waarna deze weergegeven worden op een wereldkaart — voor iedereen zichtbaar.

Hardware

Het weerstation is opgebouwd rond de ESP32-Pico-kit V4, zie het schema in **figuur 1**. Deze neemt eigenlijk alles voor zijn rekening. Om het aansluiten van de sensoren en het inbouwen in een waterdichte behuizing (een PC 100/60 HT van Fibox) wat eenvoudiger te maken, hebben we een draagprint ontworpen. De wind- en regensensoren worden op de print aangesloten via RJ45-connectoren en de BME280- en SDS011-sensoren via JST XH-connectoren. De BME280-sensor gebruikt de I²C-bus en de SDS011-sensor maakt gebruik van een UART-aansluiting. Buiten deze aansluitingen hebben we twee extra aansluitingen op de print geïmplementeerd om nog meer of andere sensoren of randapparatuur uit te kunnen lezen. De eerste extra aansluiting is een *Grove*-I²C-connector waarop Grove-modules van Seeed studio [5, 6] aangesloten kunnen worden. De tweede connector is een FTDI-connector die een UART-verbinding naar buiten uitvoert.
Als voeding neemt de schakeling genoegen met een 12V-loodaccu. Wanneer je deze accu wilt opladen met behulp van een 12V-zonnepaneel, hebben we nog iets handigs ingebouwd: de ESP32 kan een relais inschakelen als de accu leeg is en uitschakelen als de accu weer op spanning is. De spanning meet de ESP32 met behulp van een spanningsdeler met weerstandswaarden van 470 k en 100 k.

Hiermee kun je het starten en stoppen van het laden eventueel zelf nog aanpassen in de software. Led3 geeft aan of de accu opgeladen wordt of niet. Om

Figuur 1. In het schema zien we duidelijk dat de ESP32 het kloppend hart is van ons weerstation.

Figuur 2. De interne weerstand van de windvaan is afhankelijk van de windrichting.

Sensordata wereldwijd uit te lezen

het relais aan te sturen, gebruiken we een eenvoudige FET (T1) als buffer. D3 fungeert als blusdiode om de spanningspiek van de bekrachtigingsspoel van het relais af te vangen en T1 te behoeden voor een voortijdig einde. R10 beperkt de stroom naar de basis en R7 doet dat voor LED3, net als R8 en R9 voor respectievelijk LED2 en LED1.
Een DC/DC-converter (IC1) regelt de spanning terug naar 5 V. Deze converter is een stuk efficiënter dan een standaard 7805 en dus beter geschikt voor een schakeling op accu. De 5 V wordt op het ESP32 Pico-board door een stabilisator-IC van het type 1117 omgezet naar 3,3 V, die het board beschikbaar maakt voor verder gebruik via zijn aansluitpennen (hier komt de 3V3 in het schema vandaan).
D4 beschermt de ESP32 tegen een eventueel te hoge spanning afkomstig van de accu (via spanningsdeler R11/R12), terwijl C4 korte storingspieken onderdrukt en daarmee een stabiele uitlezing van de accuspanning bevordert.
R4 t/m R6 zijn pull-up-weerstanden die nodig zijn voor een goede werking van de betreffende signalen.

ONDERDELENLIJST

Weerstanden:
R1-R4 = 10 k
R5,R6 = 4k7
R7-R9 = 220 Ω
R10 = 1 k
R11 = 470 k
R12 = 100 k

Condensatoren:
C1 = 10 µF/50 V
C2 = 1 µF/10 V
C3,C4 = 100 nF

Halfgeleiders:
D1-D3 = 1N4007
D4 = BAT85
IC1 = 5V DC/DC-converter 5W, R-78E5.0-1.0 Recom Power
LED1, LED2 = led, geel, 3 mm
LED3 = led, rood, 3 mm
T1 = BS170

Diversen:
K1, K2 = 1x17-polige busstrip, steek 2,54 mm
K3 = pinheader 4-polig, steek 2,54 mm
K4 = 4-polige bus, steek 2,54 mm (Würth 691403900004B)
K5 = pinheader 6-polig, steek 2,54 mm
K6, K7 = RJ11-bus, Molex RJ11 6P V
K8 = Grove connector, verticaal, TWIG-4P-2.0-2.0
K9 = pinheader 7-polig, steek 2,54 mm, JST-XH-07-PIN-VER
RE1 = Relais ALDP105
S1 = drukknop, 6x6 mm
BME280 breakout board, I²C-versie
Wind- en neerslagmeter
Behuizing PC 100/60 HT, Fibox
Print 180468-1
Optioneel:
SDS-011 fijnstofsensor Nova Fitness

Regen- en windsensoren

De regensensor bestaat uit een klein bakje dat zich vult met regenwater. Wanneer het vol is, kantelt het en sluit dan kortstondig een magneetschakelaar die door de ESP32 gemonitord wordt. Volgens de datasheet komt één maal kantelen overeen met 0,33 mm neerslag.
De windsensor bestaat uit een windvaan die de windrichting meet en een anemometer die de windsnelheid meet. De windvaan verandert een interne weerstand van waarde, afhankelijk van de windrichting. **Figuur 2** toont de 'indeling'. Met behulp van deze windrichtingsafhankelijke weerstand en een weerstand van 10 k (R2) hebben we een spanningsdeler gemaakt. Zo creëren we een spanning die varieert wanneer de aan de windvaan gekoppelde weerstand varieert. Dit kan de ESP32 dan weer meten.
De anemometer maakt gebruik van een magneetschakelaar die twee keer per omwenteling een puls afgeeft. Volgens de datasheet is de windsnelheid in m/s gelijk aan de pulsfrequentie (in Hz) vermenigvuldigd met 0,33. Voor het bepalen van de rotatiefrequentie geeft de datasheet aan dat je het tijdverschil tussen twee pulsen in seconden met 2 dient te vermenigvuldigen (omdat de sensor twee keer per omwenteling een puls geeft), waarna je de rotatiefrequentie berekent door 1 te delen door het tijdsverschil. Vermenigvuldig je deze waarde met 0,66, dan krijg je de windsnelheid in meters per seconde.

De windsnelheid in km/u bereken je door de snelheid in m/s te vermenigvuldigen met 3,6. Oftewel: 1 rotatie per seconde komt overeen met een windsnelheid van 2,4 km/u.

De software uploaden

Om de software naar de ESP32 te kunnen uploaden, dien je de ESP32 Arduino core te installeren [7]. Daarna dien je de SPIFFS-uploadtool te installeren om de webpagina te kunnen uploaden naar het bestandssysteem van de ESP32 [8]. Als alles correct geïnstalleerd is, kun je de ESP32 pico-kit selecteren uit het boards-menu en de webpagina en sketch uploaden.

Software

Zoals gezegd, wordt het weerstation bestuurd door een ESP32. Deze configureren we met behulp van een webpagina die door de ESP32 zelf wordt gehost. Wanneer de ESP32 opstart, probeert hij verbinding te maken met het geconfigureerde netwerk. Als dit niet lukt, start de ESP32 de webserver op die de configuratiepagina host. De webserver kan ook handmatig worden gestart door de knop op de print ingedrukt te houden terwijl de voeding wordt ingeschakeld (of door de 'EN'-knop op de ESP32-module te gebruiken). Als de gebruiker de configuratiepagina 10 minuten niet heeft gebruikt, start de ESP32 opnieuw op en maakt hij verbinding met het geconfigureerde netwerk. Dit kun je onderdrukken door de knop ingedrukt te houden of continu te overbruggen.

Op de configuratiepagina kun je de actuele metingen bekijken, evenals de gemeten accuspanning. Bovendien kun je hier de netwerkinstellingen en de uploadinstellingen configureren. De netwerkinstellingen bevatten de SSID (naam) van het netwerk, die uit een keuzelijst kan worden geselecteerd, en het wachtwoord. De uploadinstellingen bevatten de benodigde API-sleutels voor Thingspeak en senseBox en de uploadperiode.

Om toegang te krijgen tot de webpagina, moet je verbinding maken met het 'ESP32'-netwerk dat de ESP32 beschikbaar maakt. De webpagina kun je dan via het IP-adres 192.168.4.1 opgeroepen. Let op dat op bijvoorbeeld Android-telefoons mobiele gegevens tijdelijk moeten worden uitgeschakeld. Dit komt doordat de telefoon vaak automatisch overschakelt naar mobiele gegevens wanneer hij geen internetverbinding via Wifi detec-

Thingspeak

Thingspeak is een gratis online database van Mathworks waarnaar gegevens kunnen worden geüpload. Hiervoor moet voor elke gemeten parameter een kanaal met een veld worden aangemaakt (**figuur 3**). Nadat het kanaal is aangemaakt, krijgt u een schrijf-API-sleutel (**figuur 4**). Deze sleutel moet op de configuratiepagina van het weerstation worden ingevoerd.

Figuur 3 Op Thingspeak maak je een nieuw kanaal aan voor je applicatie en geef je in de velden op welke data erin komt.

Figuur 4. Vervolgens krijg je een API-sleutel die je dient te gebruiken in de software van je applicatie.

teert, waardoor vervolgens de configuratiepagina niet meer bereikbaar is.
De webpagina bestaat uit één html-bestand. Dit bestand bestaat uit drie hoofddelen. Het <style>-element bovenaan definieert het uiterlijk van de pagina, het <body>-element bevat de structuur en inhoud van de pagina en het <script>-element onderaan definieert de functionaliteit van de pagina. Voor een goede werking van de webpagina moet javascript ingeschakeld zijn in de webbrowser.
De communicatie tussen de webpagina en de ESP32 gebeurt via een 'asynchrone XMLHttpRequest' op de webpagina. Wanneer de pagina wordt geladen, wordt er een http-aanvraag gegenereerd en naar de ESP32 gestuurd. De ESP32 ontvangt deze aanvraag en stuurt een antwoord terug naar de webpagina met de gevraagde instellingen of waarden. Wanneer je bijvoorbeeld op een 'submit'-knop op de webpagina drukt, wordt een gelijkaardig verzoek verzonden dat nu de door jou ingevoerde waarden bevat.

SenseBox

SenseBox is een kit waarmee je gemakkelijk je eigen 'Sensorstation' kunt inrichten. De gemeten parameters kunnen geüpload worden naar opensensemap.org, waar metingen van alle senseBoxen of met de senseBox compatibele apparaten kunnen worden bekeken. Om je eigen weerstation te registreren, maak je een account aan op openensemap.org, waarna je een senseBox kunt toevoegen. In het aanmaakmenu selecteer je het type senseBox dat je hebt, of je configureert je meetstation zelf door de gemeten parameters, gemeten eenheden en het sensortype toe te voegen (**figuur 5**). Het ESP32-weerstation moet je handmatig aanmaken. Nadat je het weerstation hebt geregistreerd, krijg je een station-ID en voor iedere sensor een sensor-ID. Deze ID's voer je in op de configuratiepagina van het ESP32-weerstation, waarna het de meetwaarden kan uploaden.

Figuur 5. Bij opensensemap.org maak je je eigen sensoren voor het ESP32-weerstation aan.

In gebruik

Nu rest alleen nog het opstellen van de hele schakeling en het aansluiten van de sensoren. Om de data te kunnen uploaden, is het natuurlijk van belang dat het (thuis)netwerk bereikbaar is voor de ESP32. En voor het opstellen van de sensoren dien je natuurlijk rekening te houden met afzonderlijke eisen van de sensoren. Een neerslagmeter vlak bij een muur is natuurlijk geen goede opstelling. Maar dat laten we graag aan uw gezond verstand over! ◀

180468-0

IN DE STORE

→ 180468-1 kale print
URL

→ 180468-71 Kit met onderdelen
URL

→ SKU 18696 Weerstation-kit
URL

Overige shopproducten:

→ SKU18423 ESP32 Pico Kit
URL

→ SKU17954 BME280 module (I^2C-versie, 160109-91)
URL

Weblinks

[1] www.inovafitness.com/en/a/chanpinzhongxin/95.html
[2] www.elektor.com/bme280-mouser-intel-i2c-version-160109-91
[3] https://thingspeak.com
[4] https://sensebox.de/en
[5] www.elektor.com/seeedstudio
[6] www.seeedstudio.com/s/grove.html
[7] https://github.com/espressif/arduino-esp32
[8] https://github.com/me-no-dev/arduino-esp32fs-plugin

De SN76477

vreemde onderdelen

Tore Bergvill (Noorwegen); redactie **Neil Gruending** (Canada)

Toen ik de kelder aan het opruimen was, kwam ik een aardige collectie documenten tegen en één daarvan was een handleiding van de SN76477-soundchip [1]. Deze was geschreven met een typemachine en had handgetekende illustraties, iets wat je vandaag de dag niet meer tegenkomt. Die SN76477 heb ik nog steeds in mijn bezit; destijds heb ik daarmee een fantastisch vibrato geluidseffect gemaakt voor een Ghostbusters Proton Pack-uitrusting.

Texas Instruments lanceerde de SN76477 Complex Sound Generator in 1978; hij werd al snel omarmd door fabrikanten van spelletjes en gamecomputers, maar ook door hobbyisten. Elektor noemde de chip een 'IC vol met verrassingen' en wijdde er in 1978 en 1981 zelfs een aantal artikelen aan, met verschillende schakelingen, een printontwerp en ontwerpsuggesties. De in **figuur 1** afgebeelde chip bevat een super-laagfrequent generator, een VCO en een ruisgenerator; je kon het geheel met een handvol passieve componenten instellen. Een mixer en een omhullende-generator combineerden deze signalen om complexe geluidseffecten op te wekken. Sommige van de voorbeeldontwerpen in de TI-manual [2] noemden de geluidseffecten Siren, Phasor Gun, Race Car, Gun Shot, Explosion, Barking Dog, Steam Train with Whistle en zelfs Bird Silencer with Explosion. De SN76477 bleek uitermate geschikt om geluidseffecten te genereren, maar minder voor toepassing in muziekinstrumenten omdat het moeilijk was om de 'overall' toonhoogte van het uitgangssignaal in te stellen.

De SN76477 werd destijds in veel commerciële producten toegepast. Diverse Taito arcade-games gebruikten de chip, hoewel Space Invaders de eerste en waarschijnlijk populairste was met het door de SN76477 gegenereerde vliegende-schotel geluidseffect. Het Zweedse bedrijf Luxor gebruikte de SN76477 in hun op de Z80 gebaseerde ABC 80 homecomputer (**figuur 2**). Luxor was een grote fabrikant van TV's en ander bruingoed in Scandinavië en hun ABC 80 was vooral in Zweden populair vanwege de grote hoeveelheid in het Zweeds geschreven Basic-software. Texas Instruments kwam al in 1980 met een nóg veelzijdiger geluidschip onder de naam SN76489 Digital Complex Sound Generator. Deze werd snel populair in game- en homecomputers zoals de Acorn BBC Micro, ColecoVision, Sega Game Gear en vele andere. De SN76477 werd in de jaren '90 uit productie genomen, maar vervangers zijn nog steeds te koop op onder andere eBay, en er is ook een modern alternatief op de markt met het typenummer ICS76477.

De manier waarop de SN76477 geluiden genereerde is inmiddels achterhaald, maar hij plaveide wel de weg voor de ontwikkeling van latere analoge geluidschips zoals de General Instrument AY-3 (Elektor TV Games Computer [3], TV Games, ZX Spectrum) en MOS Technology 6581 SID (Commodore 64) IC-families. Daarna begon het digitale tijdperk, maar dat is een ander verhaal! ◀

(160556)

Weblinks

[1] Wikipedia over de SN76477: https://en.wikipedia.org/wiki/Texas_Instruments_SN76477

[2] TI SN76477-manual: www.ece.usu.edu/ece_store/spec/SN76477.pdf

[3] Elektor TV Games Computer (Retrotronica): www.elektormagazine.com/magazine/elektor-200810/18930

1

2

Homelab Helicopter

samengesteld door **Clemens Valens** (Elektor Labs)

Met opzet niet te repareren ?
Niet waar !

Van onze speciale verslaggever
Laurent Labbe

Klanten winnen

U hebt vast al eens ergens gelezen dat fabrikanten bewust elektronische producten maken die moeilijk of zelfs helemaal niet door de eindgebruiker kunnen worden gerepareerd.
Na meer dan dertig jaar in de mobiele-telefoonindustrie, waarvan vijftien jaar verantwoordelijk voor kwaliteit en after sales-service bij een Chinees bedrijf, wil ik u graag uitleggen dat dit niet waar is en dat er bij de fabrikanten geen sprake is van enige opzet. Dit is mijn persoonlijke mening, gebaseerd op ervaringen van mijzelf en van anderen uit de wereld van de mobiele telefonie. Ik wil hiermee bijdragen aan het debat binnen de community van elektronici zoals u en ik, die dit tijdschrift lezen.

In het geval van een smartphone is een projectteam verantwoordelijk voor ontwikkeling, productie en after sales-service. Zij moeten voldoen aan alle voorwaarden voor een geslaagde marktintroductie. Voor het succes van een product zijn twee factoren bepalend: het uiterlijk en de prijs. Tien jaar geleden was iedereen nog tevreden met een telefoon die 2 cm dik was en € 200 kostte. Tegenwoordig wil de klant een apparaat dat twee keer zo dun is en de helft kost. De wereld van de mobieltjes wordt gedicteerd door prijs en uiterlijk.

Het projectteam heeft dus als opdracht om een product te creëren dat er sexy uitziet, slank, zonder schroeven, zonder klepje voor de accu (te dik) en dat zo goedkoop mogelijk is. Daarnaast doen twee leden van het team alles om ervoor te zorgen dat het product repareerbaar is: zij zijn verantwoordelijk voor de productie en voor de after sales-service. De fabriek moet tienduizenden exemplaren per dag produceren en daarnaast de exemplaren repareren die niet door de eindcontrole komen. Meestal is in de productieketen zo'n 90% van de producten in één keer goed en moet 10% op gespecialiseerde lijnen worden gerepareerd. Tot slot komen tussen 5% en 10% van de verkochte apparaten bij de after sales-service terecht. Deze kostenposten kunnen 'exploderen' als het product niet kan worden gerepareerd.

De invloed van het design op de repareerbaarheid is enorm. Hoe moet je bijvoorbeeld een touchscreen van een elegant dun apparaat repareren? Het scherm is vastgezet met lijm of dubbelzijdig tape en er is een verwarmingsplaat nodig om het los te maken. Een telefoon met een achterkant van glas ziet er leuk uit, maar is door de gemiddelde gebruiker niet te repareren omdat het glas ook is vastgelijmd. Hetzelfde onderdeel, maar dan van ABS, zou gemakkelijk kunnen worden vervangen, maar de telefoon ziet er dan minder aantrekkelijk uit en is dikker — en zal niet worden verkocht.

Veiligheid voor alles

Er is nog een andere factor om rekening mee te houden: de 'gevaarlijke' onderdelen van de telefoon (lader, accu) mogen niet gemakkelijk door een doorsneegebruiker kunnen worden gedemonteerd. U zult verbaasd zijn hoeveel mensen denken dat ze alles kunnen repareren (dat zijn er veel meer dan diegenen die het echt kunnen). In sommige landen, zoals de Verenigde Staten, geldt dat de fabrikant aansprakelijk is als een gebruiker een eenvoudige handeling kan uitvoeren met een alledaags stuk gereedschap (zoals een schroevendraaier) en er daarbij een ongeluk gebeurt. Neem bijvoorbeeld het geval van een telefoon met een LiPo-accu onder een afscherming die kan worden verwijderd. Als de afscherming met duidelijk zichtbare schroeven is bevestigd en de gebruiker

deze (ondanks de opgedrukte waarschuwingen) kan verwijderen en vervolgens met de schroevendraaier probeert de accu los te wrikken, bestaat het risico dat de accu wordt doorboord en vlam vat. Volgens de wet is in dat geval de fabrikant aansprakelijk en niet die @#$% gebruiker. Zoiets is in de Verenigde Staten al gebeurd. Dit verklaart ook waarom de behuizing van het merendeel van de netadapters ultrasoon is gelast en niet is vastgeschroefd...

De productiekosten, en daarmee de verkoopprijs (die de klant steeds lager wil hebben), hebben een directe invloed op het ontwerp van het apparaat. Zo worden bij een product aan de onderkant van het prijsspectrum kosten bespaard door de kabel van het beeldscherm vast te solderen inplaats van een connector te gebruiken, en de draden van de luidspreker vast te solderen inplaats van veercontacten te gebruiken. Deze strijd van 'altijd weer goedkoper' heeft als direct gevolg dat de apparaten lastiger te demonteren en dus lastiger te repareren zijn. En geen enkele eindgebruiker zal iets meer willen betalen voor een apparaat dat 'beter repareerbaar' is, want hij heeft er immers garantie op.

Alles op een rij

Dat apparaten bewust worden ontworpen om direct na de garantieperiode stuk te gaan is een oud gerucht dat steeds weer de kop opsteekt. Alle fabrikanten hebben echter voor ieder nieuw product een standaard-testplan. Dit plan komt voort uit jarenlange ervaring met de manier waarop je een apparaat kunt maken dat niet snel defect raakt en met een aantrekkelijke prijs. Hiervoor worden dagelijkse gebruiksprofielen opgesteld, waaruit voor de test de referentiewaarden met 1000 worden vermenigvuldigd (1000 dagen = bijna 3 jaar). Stel dat een gebruiker maximaal zes keer per dag zijn telefoon van een hoogte van 10 cm laat vallen, dan moet het apparaat 16.000 mini-valpartijen aankunnen. Dit wordt voor alle denkbare situaties gedaan, en ieder nieuw apparaat moet al deze tests doorstaan voordat het op de markt wordt gebracht. Natuurlijk, je zou ook 3650 dagen (10 jaar) als rekeneenheid kunnen gebruiken. In dat geval moeten de materialen en onderdelen robuuster zijn, en dus duurder, wat leidt tot een hogere verkoopprijs die de klant niet zal willen betalen.

Conclusie

Het zal duidelijk zijn dat de moeilijke repareerbaarheid van bepaalde apparaten een directe consequentie is van de voorkeur van consumenten voor producten met een gelikt design, en dat het absoluut geen vooropgezet plan van de fabrikanten is. Ik werk al dertig jaar in deze industrie en ik heb meegewerkt aan de ontwikkeling van honderden telefoons; nooit is er sprake geweest van het produceren van een onrepareerbaar apparaat om de klant sneller een nieuwe te laten kopen.

Dure reserve-onderdelen

Reserve-onderdelen zijn vaak duur, en de reparatie van een product na de garantieperiode is kostbaar. Dit heeft te maken met wettelijke voorschriften, industriële beperkingen en de wereldwijde markt. De wet verplicht tot reparatie onder garantie gedurende een periode van een of twee jaar na aankoop (afhankelijk van het land). Ook is het verplicht om nog gedurende drie jaar na de laatste verkoop reserve-onderdelen te leveren. De fabrikant moet dus tijdens de productie een voorraad onderdelen voor de after sales-service aanhouden die al snel kleiner zal worden. Het merendeel van de onderdelen in deze industrie heeft een erg korte levenscyclus, voornamelijk ten gevolge van technologische ontwikkelingen. Het is vaak onmogelijk ze later nog na te bestellen. Als de fabrikant een product waar garantie op zit niet kan repareren, moet hij het omruilen voor een nieuw exemplaar. Dat is puur verlies. Hij heeft er dus alle belang bij om zijn losse onderdelen zoveel mogelijk te bewaren voor reparaties onder garantie, waardoor reparaties buiten de garantie een lagere prioriteit hebben en dus duurder zullen zijn. Bovendien zullen goedkope reserve-onderdelen wereldwijd een 'grijze markt' en illegale handel veroorzaken.

Homelab-hebbedingetjes

Dit zijn twee erg handige USB-kabels: in de eerste plaats een USB-kabel met geïntegreerde schakelaar. Om uw boards of apparaten uit te schakelen of te herstarten hoeft u de kabel niet meer los te halen, maar kunt u eenvoudig op de knop drukken. Hierdoor zullen de connector en de kabel ook minder snel slijten. De andere USB-kabel heeft een extra lang connectoreinde (8 mm). Deze kabel is onmisbaar in situaties waarin het niet mogelijk is om de micro-USB-connector dicht bij de wand van de behuizing te plaatsen, of waarbij de opening te klein is voor de connector.

www.elektor.nl/usb-a-to-micro-usb-b-cable-with-switch

www.elektor.nl/8mm-micro-usb-connector-white-2m

180567-03

Wilt u meedoen? Stuur uw commentaar, suggesties, tips en trucs naar labs@elektor.com

Tips en trucs

van lezers voor lezers

Weer zo'n slimme oplossing die het elektronicaleven gemakkelijker maakt

SMD-testadapter

Dipl.-Inf. Karl-Ludwig Butte

Hebt u wel eens geprobeerd om een SMD-weerstand of een SMD-transistor door te meten met een ohmmeter? Waarschijnlijk ging het bij u net als bij mij: de weerstand sprong als een vlo van de werktafel en is sindsdien niet meer waargenomen.

De oplossing? Men neme een haarspeld (**figuur 1**), een stukje gaatjesprint, drie printpennen, een stukje koperlakdraad, en vervaardige daaruit in een handomdraai een kleine maar zeer nuttige SMD-testadapter.
Schuif als eerste de haarspeld in de richting van de koperbanen op de gaatjesprint. Aan het eind van de haarspeld onderbreekt u vervolgens de koperbaan bij het eerstvolgende gat met behulp van een horlogemakers-schroevendraaier of iets dergelijks (**figuur 2**). Vervolgens soldeert u twee korte draadbruggen van koperlakdraad en de drie printpennen zoals in **figuur 3**. Het gat juist boven de doorgesneden printbaan vult u met wat soldeertin. In **figuur 4** en **figuur 5** ziet u hoe straks een SMD-weerstand en een SMD-transistor op onze meethulp worden geplaatst. Maar eerst moet u nog de haarspeld monteren. Schuif hem op de print zoals in **figuur 6** en zorg ervoor dat het eind van de haarspeld op het gat drukt waar de printbaan onderbroken is. Draai nu de print om en lijm de haar-

Hebt u zelf een slimme oplossing voor iets dat echt onhandig is? Gebruikt u gereedschap of een component op een ongewone manier? Hebt u een idee hoe een probleem gemakkelijker of beter aangepakt kan worden? Laat het ons weten — voor elke tip die wij van u publiceren ontvangt u 40 euro!

speld vast met behulp van een lijmpistool (**figuur 7**). Door de mechanische spanning staat de haarspeld iets van de print af; daarom moet u hem met een geschikte tang op de print drukken totdat de lijm is uitgehard.

Tijd om onze SMD-test-adapter uit te proberen! Schuif een SMD-weerstand onder de haarspeld en sluit een ohmmeter aan op de desbetreffende printpennen (**figuur 8**). Op de bovenste schaal van mijn multimeter lees ik dan een waarde van ruim 1 k af, wat bij dit meetbereik neerkomt op 1 MΩ. De opdruk moeten we dus lezen als “105” en niet, zoals je ook zou kunnen denken, als “501”, wat 500 Ω zou betekenen. Wat fijn dat we dat nu gewoon even kunnen nameten!

Ten slotte nog een test met een transistor. Schuif een SMD-transistor onder de haarspeld zoals in figuur 5 en sluit een transistortester aan (**figuur 9**). De tester geeft aan (a) dat de transistor in orde is, (b) dat het een PNP-type is en (c) dat de zwarte krokodillenklem op de basis van de transistor rust. ◀

170569-03

Snore Shield

Helpt je nachtrust te verbeteren

Clemens Valens (Elektor Labs)

Veel mensen snurken en velen vinden het niet erg of lijken het lawaai niet op te merken. Helaas lijdt een ander groot deel van de mensheid aan slaapgebrek als gevolg van het snurken van bedpartners en/of kamergenoten. Dit project is voor die laatste categorie.

Helpt je nachtrust te verbeteren

Veel mensen snurken en velen vinden het niet erg of lijken het lawaai niet op te merken. Helaas lijdt een ander groot deel van de mensheid aan slaapgebrek als gevolg van het snurken van bedpartners en/of kamergenoten. Dit project is voor die laatste categorie.

Snurkstress

Het probleem bij het proberen in slaap te vallen in dezelfde kamer als een snurker is dat zowel het focussen als het anticiperen op de snurkende geluiden stress veroorzaakt, waardoor je wakker blijft. Schoppen, slaan, duwen of porren van de snurker helpt niet altijd en verhoogt doorgaans de hoeveelheid stress, waardoor het nog moeilijker wordt om in slaap te vallen.

Het Snore Shield

Het hier gepresenteerde apparaat produceert een aangenaam geluid voor de niet-snurker om naar te luisteren en hopelijk te ontspannen. Het apparaat "luistert" ook naar gesnurk. Wanneer het snurken detecteert, verhoogt het het volume van het ontspannende geluid om het snurken te maskeren. Na vijf minuten - hopelijk genoeg tijd om het snurken te laten stoppen of voor de niet-snurker om in slaap te vallen - neemt het geluidsniveau af, voltooit het een cyclus en begint snurkdetectie opnieuw.

Ontspannend geluid

De geluiden van de zee, een lichte branding en wind door de bomen worden door veel mensen als ontspannend geluid ervaren. Het Snore Shield (**figuur 1**) produceert daarom witte ruis (met instelbare kleur), wat in de buurt komt van deze ontspannende geluiden. Wilt u andere geluiden horen zoals stromend water met

Het Snore Shield maakt van elke slaapkamer een hightech slaaplaboratorium (in tegenstelling tot Elektor Labs)

PROJECT INFO

Gezondheid
Arduino

beginners
➡ gevorderden
experts

ongeveer 2 uur

SMD-soldeergereedschap, Arduino IDE

€25

ontspannende muziek, dan kunt u een MP3-speler module aansluiten.

Het begon als een Android-app

Het Snore Shield is gebaseerd op de Android-applicatie 'SleepSation' (sic; geen typo; beschikbaar in de Google Play Store). Omdat deze app op een mobiele telefoon (of tablet) werkt, moet je een telefoon naast je bed houden. Helaas willen veel mensen geen mobiele telefoon naast hun kussen, daarom hebben de ontwikkelaars van de app ons gevraagd een smartphone-vrij apparaat te ontwerpen.

Het Snore Shield voldoet aan de specificaties voor een Arduino-shield en kan op een Arduino worden aangesloten, bij voorkeur een Uno (om het klein te houden), maar andere typen zouden ook moeten werken.

Twee modi

Het apparaat heeft twee bedieningsmodi:

- continu geluid;
- snurkgerelateerd geluid.

In de modus continu-geluid is het ontspannende geluid continu aan. Wanneer snurken wordt gedetecteerd, wordt het geluidsniveau gedurende vijf minuten verhoogd.
In de modus snurkgerelateerd is het ontspannende geluid alleen ingeschakeld wanneer snurken wordt gedetecteerd. Drukknop SW2 schakelt tussen de twee modi.
Door op drukknop SW1 te drukken, wordt de snurkdetectie geannuleerd als er gesnurkt werd en het geluidsniveau keert terug naar het standaardniveau (of uit of laag, afhankelijk van de bedieningsmodus). Als u er op drukt wanneer hij inactief is, wordt een snurkdetectie geforceerd, waardoor het geluidsniveau toeneemt tot snurkmaskerend niveau.

Snurkdetectie

Het Snore Shield volgt continu het omgevingsgeluidsniveau. Onder normale slaapomstandigheden moet dit niveau laag zijn. Snurken verhoogt het gemiddelde geluidsniveau, maar dat geldt ook voor andere geluidsbronnen, zoals bewegen in bed of passerend verkeer. Het detecteren van snurken is daarom een beetje gecompliceerder dan het vergelijken van het huidige geluidsniveau met een lopend gemiddelde.
Het Snore Shield vereist drie snurken op een rij, waarbij een snurk wordt gedefinieerd als een kort (tot ongeveer drie seconden) hard geluid gevolgd door een periode van stilte van vijf tot tien secon-

Figuur 1. Het Snore Shield past op een Arduino Uno.

Figuur 2. Het schema van het Snore Shield bestaat voornamelijk uit microfoonversterker T1, spanningsgeregelde ruisprocessor T2 & T3 en eindversterker IC1.

Specificaties

- Twee snurkdetectie modi
- Electret microfoon
- Koptelefoon uitgang
- OLED I²C grafisch display
- Twee drukknoppen
- Volume potentiometer
- Regeling van de ruiskleur
- Compatibel met Grove MP3-speler module
- Voeding met 9-12 VDC-adapter of 5V USB-powerbank

den. Natuurlijk kunnen valse detecties niet worden uitgesloten en geluiden met vergelijkbare kenmerken kunnen het apparaat ook activeren, zoals drie opeenvolgende (langzame) hoestbuien.

Details over de innerlijke werking

Het schema van het Snore Shield is afgebeeld in **figuur 2**. Snurkdetectie begint bij de microfoon, MIC1, aangesloten op een versterker met een doorlaatband van ongeveer 100 Hz tot 3 kHz. De uitgang van de versterker is verbonden met analoge ingang A0 van de Arduino ATmega328 die zorgt voor digitalisering en digitale verwerking. Binnen in de MCU wordt het audiosignaal gesampled met 7 kHz. Hoewel dit meer dan twee keer de ingangsbandbreedte lijkt, is de helling van de versterker in werkelijkheid niet erg steil en zal de frequentieanalyse van het audiosignaal moeilijk zijn. Het is hier geen probleem, we kijken gewoon naar het geluidsvolume.

Om de snurkende geluiden te maskeren, produceert het systeem witte ruis. Dit gebeurt in software met behulp van een lineair feedback schuifregister (LFSR) met behulp van een 32-bits schuifregister en een bemonsteringsfrequentie van 20 kHz. De digitale ruis komt via pen D8 van de Arduino naar buiten.

Transistor T2 functioneert als een spanningsgestuurde versterker om amplitudemodulatie van het ruissignaal mogelijk te maken. De stuurspanning wordt geproduceerd door een lowpass-gefilterd PWM-signaal dat beschikbaar is op Arduino pen D6. De schakeling rond transistor T3 is een filter om de ruis “kleur” aan te passen. Voor de eenvoud wordt de feedback van het filter geregeld met een klassieke potentiometer. De software kan transistor T4 aan- of uitschakelen om tussen twee feedbackmodi te kiezen, wat resulteert in twee soorten ruis.

De uitgang van de ruisversterker is verbonden met een klassieke LM386 eindversterker met handmatige volumeregeling. Deze versterker kan een koptelefoon of kleine luidsprekers aansturen.

MP3-speler voor eigen geluiden

De gebruikte MP3-speler module is een Grove-module van Seeedstudio (**figuur 3**) en wordt aangesloten op

Figuur 3. Er kan een MP3-speler module worden aangesloten om de algehele geluidskwaliteit te verbeteren en om het Snore Shield aangepaste geluiden te laten spelen.

Gebruik een luidspreker van goede kwaliteit

Het is belangrijk dat het ontspannende geluid ook echt ontspannend is, dus zorg ervoor dat u een goed geluidssysteem aansluit. Een waardeloze luidspreker produceert waardeloos geluid en verknalt het doel van het systeem. Lees het gedeelte over de voeding elders in dit artikel aandachtig door. Een volumeregeling en regeling van de geluidskleur zijn beschikbaar om het geluid aan te passen.

K7. Deze module wordt bestuurd via een seriële poort die is geëmuleerd in software op Arduino-pinnen D2 en D3. Andere typen MP3-speler modules die op een soortgelijke wijze worden bestuurd kunnen ook worden gebruikt als u de software zelf wilt aanpassen. Merk op dat Grove-connector K8 een speciaal type is met een steekvan 2 mm. K7 biedt hier een 2,5mm-alternatief.
De seriële poort van de software heeft helaas een negatieve invloed op het digitale ruissignaal omdat het interrupts tijdens het verzenden uitschakelt, waardoor er hoorbare storingen in het interrupt-gestuurde ruissignaal worden geïntroduceerd. Als de MP3-module niet wordt gebruikt, moet u een regel in het programma commentaar maken om deze problemen te onderdrukken. Een commentaar in de code geeft aan om welke regel het gaat.

Het display

Het OLED-display is een van die populaire 0,96 “64 × 128-pixel grafische types. Deze schermen bestaan in verschillende versies, met zes of vier pennen en met verschillende pennen; we kozen voor het 4-pens type. Dit type is er in twee smaken: pen 1 = GND, pen 2 = VCC en andersom. Met soldeerjumpers JP1 en JP2 kunt u een van beide kiezen.
Een display is praktisch voor interactie met de gebruiker, maar op dit moment is het niet erg behulpzaam. Het is toegevoegd omdat de SoundSation-app een aantal timer- en opname-opties heeft die (nog?) niet zijn geïmplementeerd in het Snore Shield.

Figuur 4: Het OLED-scherm van het Snore Shield geeft in eerste instantie nutteloze debug-informatie. “298” op de 4e regel is het aantal seconden voordat het systeem weer in snurkdetectie-mode gaat (hier was het handmatig getriggerd).

Op de eerste regel (**figuur 4**) toont het de naam van het apparaat (voor het geval u het bent vergeten) en de firmwareversie; op de tweede is het Elektor-projectnummer te zien.
De derde regel toont geluidsstatistieken (detectiedrempel en omgevingsge-

Advertentie

Win...

En help ieder huishouden aan een veiliger huis!

Help de Nederlandse Brandwonden Stichting en bedenk een akoestisch/visueel waarschuwingssysteem dat zoveel mogelijk passief werkt. Ontwerp een schakeling die de temperatuur van een apparaat in de gaten houdt en die tegelijk zelf zo weinig mogelijk (of helemaal geen) energie gebruikt. De schakeling moet alarm slaan, akoestisch en/of visueel, wanneer er kans is op brandgevaar.

Het winnende ontwerp wordt beloond met een publicatie in Elektor. Daarnaast gaan we met uw schakeling aan de slag en brengen we hem op de markt als product waar u royalties over ontvangt.

Stuur uw idee vóór 1 maart 2019 naar NBS-wedstrijd@elektor.nl en wordt beroemd!

nederlandse brandwonden stichting

Steun ons via www.elektormagazine.nl/brandwondenstichting

luidsniveau in dB) en het geluidsniveau (als een percentage). Aan het einde is de modus: ‘0’ voor continu geluid of ‘1’ voor snurkgestuurd geluid. Aan het einde is het ruistype ‘0’ of ‘1’.
Ten slotte toont de vierde regel de snurk-detectiestatus met een aantal cijfers die wat rond tellen. Wanneer het laatste cijfer tot drie telt, gaat het systeem naar de snurkmaskerende geluidsmodus. Snurkdetectie stopt en deze getallen blijven gedurende vijf minuten staan. Het volumeniveau op regel drie wordt langzaam verhoogd naar een hoger percentage.

Software

De software die nodig is om het Snore Shield operationeel te krijgen, kan gratis worden gedownload van [1], en u wordt aangemoedigd om ermee te experimenteren. Misschien kunt u het algoritme voor snurkdetectie verbeteren of wilt u misschien de geluidsniveaus aanpassen? Het is een Arduino-sketch in één bestand en een reeks bibliotheken en kan als zodanig zonder speciaal gereedschap in het board worden geprogrammeerd. Het enige wat je nodig hebt is een recente Arduino IDE (wij gebruikten 1.8.0).
Deel uw bevindingen alstublieft met de wereld door ze te plaatsen op [1].

180481-02

De voeding

Een typische stand-alone Arduino-applicatie wordt gevoed vanuit een telefoonoplader. Helaas zijn deze laders elektrisch gezien erg luidruchtig, wat de kwaliteit van het ontspannende geluid van het Snore Shield ernstig beïnvloedt. Het wordt daarom sterk aanbevolen om het apparaat van een ouderwetse 9 VDC (12 VDC max) netstekervoeding (voedingsadapter) te voeden, hoewel een USB-powerbank ook goed zou moeten zijn. Dus: gebruik geen telefoonoplader of een USB-poort van een PC.

ONDERDELENLIJST

Weerstanden
(SMD 0805, tenzij anders vermeld)
JP1,JP2 = 0 Ω
R15,R18 = 10 Ω
R6 = 680 Ω
R13,R16 = 1 k
R4 = 1k2
R3 = 2k7
R5,R7,R8,R9,R12 = 10 k
R2 = 15 k
R1 = 27 k
R14,R17 = 100 k
R10,R11 = 180 k
P2 = 10 k, potentiometer, verticaal
P1 = 100 k, potentiometer, verticaal

Condensatoren
(SMD 0805, tenzij anders vermeld)
C5,C6 = 100 pF
C7 = 1 nF
C9 = 47 nF
C2,C14,C15,C16 = 100 nF
C1,C8 = 1 µF
C10,C12 = 10 µF / 16 V, steek 2 mm
C3,C4,C13 = 47 µF / 16 V, steek 2,5 mm
C11 = 220 µF / 16 V, steek 5 mm

Halfgeleiders
T4 = 2N7002
T1, T2, T3 = BC847C
IC1 = LM386M-1/NOPB

Diversen
MIC1 = electret capacitieve microfoon
K6 = jack connector, 3,5 mm
S1,S2 = drukknop, 6x6 mm
K1 = 6-pens header, steek 2,5 mm
K2,K3 = 8-pens header, steek 2,5 mm
K4 = 10-pens header, steek 2,5 mm
K5 = 4-voudige bus, steek 2,5 mm
K7 = 4-pens header, steek 2,5 mm
K8 = 4-pens header, steek 2 mm
MOD1 = OLED display, 0.96", I²C
print #180481-1

@ WWW.ELEKTOR.COM

→ **180481-1 Snore Shield, kale print**
www.elektor.nl/snore-shield-180481-1

→ **SKU18668: Grove MP3 v2 – Programmeerbare compacte audio speler**
www.elektor.nl/grove-mp3-v2

→ **SKU18671: Grove universele 4-pen connector (steek 2 mm)**
www.elektor.nl/10-x-4-pin-2-mm-pitch

Weblink

[1] Project op Elektor Labs: www.elektormagazine.nl/labs/snore-shield

Speed-o-meter

Met tripmeter en logger

Bera Somnath (India)

Voor het bijhouden van de afgelegde weg en het berekenen van de gemiddelde snelheid tijdens het joggen heeft de auteur een Arduino Nano gekoppeld aan een eenvoudige GPS-ontvanger en een OLED-display. Dankzij de modulaire aanpak past de schakeling gemakkelijk in je hand en met wat extra moeite is er zelfs een sleutelhangerschakeling van te maken.

’s Ochtends ga ik vaak joggen en al geruime tijd probeer ik mijn snelheid op een redelijk betrouwbare manier te meten. Ik heb het in eerste instantie met een hoogfrequente Dopplersensor geprobeerd, maar daarmee lukte het niet om de eigen snelheid te meten. Wat er wel mee lukte, was de relatieve snelheid meten op het moment dat er een groot voorwerp langs de zijkant passeerde. Daarna heb ik geëxperimenteerd met een BME280-sensor. Deze sensor meet atmosferische druk, temperatuur en luchtvochtigheid. Mijn gedachte was dat een verandering in snelheid de luchtdruk in de horizontale richting zou veranderen en dat een loodrecht op deze richting gemonteerde sensor dit zou kunnen meten. Maar dat was helaas niet het geval. Of in ieder geval was het luchtdrukverschil te klein om te meten met de BME280.

Figuur 1. In het schema zien we door het gebruik van modules slechts 7 onderdelen, aan elkaar geknoopt door de ATmega328P.

Voor mijn volgende (en laatste) poging heb ik een GPS-ontvanger[1] ingezet. Deze slaagde er gekoppeld aan een Arduino Nano gelukkig wél in om, wanneer er ten minste drie satellieten binnen zicht waren, een voldoende nauwkeurige snelheid te kunnen meten. Een bijkomstig voordeel van een GPS-ontvanger is dat ik er naast de snelheid ook de momentane tijd en afgelegde weg van mijn jogging-exercitie mee kan meten. En van daaruit kan ik ook de in totaal afgelegde weg en de gemiddelde snelheid tijdens een *run* bepalen. Een leuk OLED-display erbij en het geheel is compleet.

Meetfilosofie — V1

Veronderstel een situatie wanneer je tijdens het lopen terugkeert en dezelfde weg terug loopt. De afgelegde afstand, die een functie is van de breedtegraad en lengtegraad van de afgelegde weg, zal in dat geval nul zijn, maar feitelijk heb je de weg twee keer afgelegd. Om dit probleem aan te pakken, heb ik een drukknop (‘P’, S3) op digitale pen 6 van de microcontroller aangesloten (zie **figuur 1**). Deze dien je in te drukken op het moment dat je begint te lopen en daarna telkens als de weg draait of afbuigt en niet (meer) recht is.

Stel, je jogt van A naar B naar C en komt uiteindelijk weer terug naar A (zie

Figuur 2. Een jogger die van A naar B naar C en weer naar A rent, is uiteindelijk niet van plaats veranderd, maar heeft wel een afstand afgelegd.

Waarom geen smartphone?

Mijn dochter vroeg mij waarom ik hier niet gewoon een smartphone voor gebruikte — een legitieme vraag van de nieuwe generatie. Ik heb er zo mijn redenen voor...

- Ik hou er niet van om mijn telefoon mee te nemen tijdens het joggen. Het leidt af.
- Met 'jogging' vind je meer dan 30 apps variërend tussen 7 en 181 MB. Ik zie door de bomen het bos niet meer.
- In sommige apps moet je de nodige gegevens over jezelf invoeren: stapgrootte, lichaamsgewicht, lengte, ... Ik ren dagelijks gewoon 6 á 7 km met 9-10 km/u en relax daarna. Ik heb geen behoefte aan al die andere mogelijkheden.
- Bijna alle apps hebben een actieve internetverbinding nodig.
- Dit is een kleine schakeling die gemakkelijk in je hand past. Het kan zelfs als sleutelhanger geconstrueerd worden wanneer je de Nano vervangt door de kale SMD-ATmega328.
- Een smartphone kost minstens €150; deze schakeling kost veel minder!

Figuur 2). De totale verplaatsing is nul, maar je hebt wel een afstand afgelegd, namelijk AB+BC+CA. Om de totale afgelegde weg bij te houden, begin je met een druk op de knop P op punt A (dit zal de berekening starten), dan weer op punt B, dan op punt C, en ten slotte na aankomst weer op punt A. De totale afgelegde weg, dus AB+BC+CA ,wordt samen met de totale tijd in seconden die je nodig hebt gehad om de afstand af te leggen en het tijdstip opgeslagen in het EEPROM-geheugen van de ATMEGA328. Deze heeft hiervoor 1024 bytes ter beschikking.

Display

Wanneer P of Q niet ingedrukt zijn, toont het display van links boven naar rechtsonder de breedtegraad, de lengtegraad, het aantal satellieten waarvan de GPS-module signaal ontvangt, de huidige tijd en de momentane snelheid (in km/u), zie **figuur 3**. Wanneer P voor de eerste maal gedrukt wordt, toont het display de breedtegraad, de lengtegraad, het aantal 'zichtbare' satellieten, de huidige tijd, de huidige snelheid en "I/L:AB/AB 00 00"; 'I' staat voor 'Instant distance' (huidige afstand van het deeltraject), 'L' voor de totale afstand (cumulatief), de eerste 'AB' staat voor het aantal meters van het huidige deeltraject, de tweede 'AB' voor het aantal meter van het in totaal tot nu toe afgelegde traject, de eerste '00' staat voor het aantal deeltrajecten en de tweede '00' voor het aantal verstreken seconden sinds de start.
Wanneer P voor de tweede maal ingedrukt wordt, toont het display breedtegraad, lengtegraad, aantal zichtbare satellieten, huidige tijd, huidige snelheid en bijvoorbeeld I/L: 'CA/(AB+BC+CA)' 03 1950, waarbij CA en (AB+BC+CA) in meter worden weergegeven, '03' het aantal deeltrajecten aangeeft en '1950' het aantal seconden toont dat sinds de start van de meting verstreken is. Met deze informatie kunnen we bijvoorbeeld in **figuur 4** aflezen dat er een afstand van 12 m is afgelegd in het huidige traject, dat er 25 m in totaal is afgelegd in 5 verschillende trajecten en dat de meting 62 s geleden is gestart.
Druk je de tweede knop in ('Q', S4), dan wordt de totale tijd sinds de start van de meting getoond (in s), de afgelegde weg (in m), de actuele tijd en de gemiddelde snelheid (in km/u) gemeten over de hele afstand (zie **figuur 5**). Je kunt deze knop zo vaak indrukken als je wilt om de gegevens op het scherm weer te geven. De gegevens blijven ook na het uitschakelen van de stroomvoorziening (via S1) in het geheugen bewaard. Echter, zodra knop P wordt ingedrukt, worden deze gegevens vervangen door nieuwe gegevens.

Aanpassingen — V2

Omdat enkele vrienden van me de schakeling ook gebruikten bij hun dagelijkse trainingsroutine en ze me vroegen een paar dingen aan te passen, heb ik dat voor ze gedaan. Deze versie is hier getoond in het schema van figuur 1.

Figuur 3. Normale toestand

Figuur 4. Knop P ingedrukt

Figuur 5. Knop Q ingedrukt

Figuur 6. Knop P ingedrukt, aangepaste versie.

Tijdens de normale weergave op het scherm (figuur 4) wanneer schakelaar P eenmaal wordt ingedrukt, wordt de meter op nul gezet en begint de meter te tellen zodra hij beweegt. Op de laatste regel van het display heb ik aan de rechter kant een eenvoudige accu-indicatie toegevoegd. Zolang de spanning van de LiPo-batterij meer dan 3,3 V is, verschijnt 'B+', zoals in **figuur 6**. Eenmaal onder de 3,3 V verandert het naar 'B–', wat aangeeft dat de LiPo geladen dient te worden.
Voor het monitoren van de accuspanning heb ik een spanningsdeler toegevoegd (R1, R2). Ook heb ik de HT7333-1 spanningsstabilisator ingezet die hier heel geschikt is vanwege zijn lage spanningsval en laag stroomverbruik in rust.

Constructie en testen

De constructie is een van de eenvoudigste onderdelen van dit project. Zoek een klein stukje dubbelzijdige print en plaats de Arduino, het OLED-display en de GPS-ontvanger aan de voorzijde. De antenne van de GPS-module moet aan de voorzijde (onafgeschermd door de PCB) worden geplaatst om de beste ontvangst van GPS-satellietsignalen te waarborgen. De LiPo-cel kan aan de achterkant van de print worden bevestigd. **Figuur 7** toont mijn opgebouwde schakeling. Vergeet niet de Arduino te programmeren met de software [2].
Het testen van mijn eigen bouwsels doet mij altijd het meeste plezier. Voorzie de schakeling van de juiste spanning middels een LiPo- of Li-ion-batterij en sluit S1 en S2 (bij een grotere voedingsspanning zoals 9 of 12 V kan S2 waarschijnlijk geopend blijven). Neem de schakeling mee naar je training en eventueel je partner of vriend(in) om indruk op hem/haar te maken met je nieuwe gadget! Zodra de GPS-ontvanger voldoende GPS-satellieten ziet, zal de PPS-led (Pulse Per Signal-led) gaan knipperen en toont het OLED-display de informatie zoals hierboven besproken. ◀

180432-01

Figuur 7. De schakeling kan eenvoudig op een stukje print worden opgebouwd en is klein genoeg om in de hand te houden.

IN DE STORE

→ **Boek Mastering Microcontrollers Helped by Arduino (3e Editie):**
www.elektor.nl/mastering-microcontrollers-helped-by-arduino-edition-3

→ **GPS-module:**
www.elektor.nl/gy-neo6mv2

→ **Arduino Nano:**
www.elektor.nl/arduino-nano-3

Weblinks

[1] GPS-module: www.elektor.nl/gy-neo6mv2
[2] Downloads: www.elektormagazine.nl/180432-01

In memoriam

De auteur draagt dit artikel op aan zijn moeder, Sefali Bera, wiens overlijden hem en de gemeenschap waar zij leefde in Howrah, India diep heeft geraakt. Zij was voor hem een grote inspiratiebron voor zijn artikelen.

One-time Pad Crypto Shield

veilige communicatie via een open systeem

Luka Matić

Er is heel wat cryptografische hardware en software verkrijgbaar, dus het ligt misschien niet voor de hand om daar een eigen systeem voor te bouwen. Maar toch is er een heel eenvoudige reden om dat te doen: je kunt zulke apparaten en programma's niet vertrouwen, omdat je in 99% van de gevallen geen idee hebt van wat er binnenin gebeurt!

We beginnen dit artikel met een stukje cryptografiegeschiedenis (tip: kijk eerst naar video [1]). De bekende Duitse codeermachine Enigma uit de vorige eeuw (**figuur 1**) was een vrij eenvoudig elektromechanisch apparaat, dat bestond uit een toetsenbord, een batterij, enkele lampen en een constructie van draaischakelaars. Het was gemakkelijk om het totale aantal mogelijke combinaties (en dus de coderingssterkte) te berekenen. En ook het controleren van de juiste werking was geen probleem, omdat er alleen hardware was. Er konden dus gegarandeerd geen softwarefouten in zitten. Hij was krachtig genoeg om alle pogingen tot crypto-analyse en andere in die tijd bekende aanvalsmethoden te weerstaan. De Enigma had geen ingebouwde communicatiehardware; een coderingsofficier zorgde voor het bedienen van het toetsenbord en het aflezen van de lampen en een telegrafist zorgde voor het draadloos verzenden en ontvangen van

de berichten. Verder:

- Was het onmogelijk een *Trojan* te installeren en leesbare tekst door te sturen naar een IP-adres.
- Was overlopen van het geheugen niet mogelijk en konden toetsaanslagen niet worden afgekeken met een keylogger (hard- of software).
- Werkte hij op lage frequenties en was er bijna geen TEMPEST-straling [2]. Bovendien werd hij vaak gebruikt aan boord van onderzeeërs, waarvan de metalen romp werkte als een kooi van Faraday.

Tegenwoordig wordt de meeste versleuteling gedaan door software op een gewone PC, en voor de rest met speciale, kostbare apparatuur zoals cryptofoons. Zelfs het beste versleutelingsalgoritme is nutteloos als het draait op een onveilig apparaat zoals een PC. De meeste cryptografische hardware wordt geleverd als een black box en is dus moeilijk te analysen voor een niet-geïnformeerde gebruiker. Zelfs zonder hard- of softwarebeveiliging zijn de systemen trouwens vaak te ingewikkeld voor een grondige analyse. Het is gemakkelijk te begrijpen dat de oude Enigma het qua veiligheid op bijna alle fronten wint van deze moderne oplossingen! Begrijpt u waar ik heen wil? Een systeem is eigenlijk veiliger als het gemakkelijk is te analyseren. Als u denkt dat ik overdrijf, lees dan [3] en let vooral op het hoofdstuk over de "analyse van *attack trees*".

One-Time Pad of OTP-encryptie

Volgens Wikibooks [4] is One-Time Pad (OTP)-encryptie de enige potentieel onkraakbare manier van versleuteling. Potentieel, want het werkt alleen als de gebruikers nauwgezet volgens de regels werken.

Bij OTP wordt elk bericht versleuteld met een unieke, toevallige sleutel (zie **figuur 2**). Die sleutel moet voldoen aan de volgende voorwaarden:

- Hij moet worden gegenereerd met een niet-deterministisch, niet-herhaalbaar proces. Er mag geen algoritme aan te pas komen;
- Hij is voor éénmalig gebruik;
- Hij mag niet in vijandelijke handen komen.

OTP-encryptie is op zich gemakkelijk uit te voeren: genereer een geschikte sleutel en XOR die met de te coderen tekst. Decoderen is ook gemakkelijk: XOR het versleutelde bericht met dezelfde sleutel en de tekst is weer leesbaar. Het versleutelde bericht is dus even lang als het origineel. Vanwege de volkomen toevallige sleutel is het onmogelijk om het originele bericht af te leiden uit alleen het versleutelde bericht, want alle mogelijke teksten van dezelfde lengte zijn even waarschijnlijk. OTP is het enige cryptosysteem waarvoor dit is aangetoond.

Een praktisch systeem

OTP is dus gebaseerd op een eenvoudig, gemakkelijk te programmeren wiskundig algoritme, perfect voor een microcontroller (µC) met een bekende (Harvard) architectuur. Als de broncode van het algoritme wordt gepubliceerd, kan die in detail worden bekeken, zodat iedereen kan controleren wat er gebeurt in de µC. Het algoritme gebruikt alleen twee databuffers in RAM, die natuurlijk niet mogen overlopen.

Voor een goede sleutel is een True Random Number Generator nodig (TRNG, zie [5])

PROJECT INFORMATION

Cryptografie | OTP | TRNG | Secure Communication | Arduino

beginners
➡ gevorderden
experts

Ca. 2 uur.

SMD-soldeergereedschap, AVR-programmer, Atmel Studio

€25

Figuur 1: Een close-up van de beroemde elektromechanische codeermachine Enigma.

Figuur 2: Hoe Alice een bericht naar Bob stuurt met behulp van OTP-versleuteling.

Figuur 3: Het schema van het OTP Crypto Shield is te verdelen in drie stukken: De RS232- en Ethernet-poort rondom IC2, K5 en MOD1; de audio-interface met FSK-modem IC3 en het micro-SD-kaartslot voor het vastleggen van de coderingssleutel en de berichten.

die ook is opgebouwd uit open source-software en open hardware zodat we de werking volledig kunnen analyseren. De OTP-sleutels worden opgeslagen op een SD-kaart, zodat we gigabytes aan toevallige getallen kunnen gebruiken (geen grote papieren codeboeken meer zoals vroeger). We gebruiken een terminal die via bedraad Ethernet of seriële poort communiceert met de µC (een goedkope 7-inch-laptop die nooit met het internet mag worden verbonden, zonder wifi-module of camera, is prima). Een SD-kaartslot is natuurlijk ook vereist. Het versleutelen gebeurt in de microcontroller die, net als de Enigma, fysiek gescheiden is van andere computeractiviteiten en draadloze communicatie. Dat maakt het systeem moeilijk aan te vallen met virussen of *Trojans*. De TEMPEST-straling is natuurlijk wel veel groter dan die van de Enigma.

Het systeem is opgebouwd met goed verkrijgbare componenten zonder bijzondere hardware, dus het is gemakkelijk na te bouwen en volledig te analyseren.

Twee manieren van werken

Het OTP-systeem is op twee manieren te gebruiken: on- en offline. In online modus worden versleutelde berichten verzonden en ontvangen via een audio-verbinding. In offline-modus worden berichten en files versleuteld en opgeslagen op een medium (bv. SD-kaart) en dan naar de bestemming overgebracht met behulp van email, post of koeriersdienst.

Beschrijving van de hardware

Ons OTP-cryptografiesysteem bestaat uit een Arduino-compatibel board met het OTP Crypto Shield erop gestoken. De Elektor Uno R4 fungeert als basisboard omdat hij beschikt over een ATmega328PB-microcontroller met twee SPI-poorten en twee UART's. Het schema van het shield is te zien in **figuur 3**.

De belangrijkste onderdelen op het OTP Crypto Shield zijn:

1. Een micro-SD-kaartslot voor het schrijven van OTP-sleutelfiles en de te versleutelen/decoderen files.
2. Een RS232-poort (MAX3232) voor het verbinden van het shield met een terminal. De seriële data loopt via een echte RS232-verbinding (K5) of via Ethernet met een WIZ107SR Ethernet/RS232-adapter (mocht de terminal geen seriële poort hebben, zoals onze goedkope 7-inch laptop). Met R6 en R7 kan de Ethernet-module worden afgekoppeld als hij op het shield is gesoldeerd.
3. Een FSK-modem (TCM3105) om gecodeerde teksten via analoge audio te verzenden en te ontvangen (online-modus).

Soft shutdown

Om te voorkomen dat dezelfde OTP-reeks tweemaal wordt gebruikt, is het belangrijk te onthouden tot welke positie de pointer naar de OTP-sleutel is gekomen. Ook enkele andere belangrijke variabelen moeten worden bewaard. Als onverwacht de voedingsspanning wegvalt, moet die data snel worden opgeslagen op de SD-kaart. In dat geval zorgt supercondensator C11 ervoor dat de schakeling nog een paar seconden kan doorwerken, zo wordt een 'soft shutdown' mogelijk. De interne comparator van de µC detecteert het wegvallen van de spanning op PD6, waarna de soft shutdown-functie wordt uitgevoerd in een interrupt service routine. S3 activeert de soft shutdown handmatig.

Als de spanning wegvalt, schakelt T2 de WIZ107SR-module (indien aanwezig) uit. Die trekt meer dan 200 mA, waardoor de spanning te snel zou inzakken vanwege de interne weerstand van supercap C11. Het schrijven naar de SD-kaart zou anders onmogelijk worden.

S2 is een harde reset-knop waar met zorg mee moet worden omgegaan om te voorkomen dat essentiële data (zoals de waarde van de pointer naar de OTP-sleutel) verloren gaat. S1 zorgt voor een veilige zachte reset zonder dataverlies en wordt meestal gebruikt als de UART's vol raken in online-modus of om het zoeken naar de sleutelpointer te stoppen als dat te lang duurt.

Voedingsperikelen

De 5V-voedingsspanning wordt op K10 aangesloten. De micro-USB-connector op de Elektor Uno R4 is te gebruiken bij het flashen van de firmware met behulp van de bootloader (JP1 op het shield open), maar omdat het invloed kan hebben op het resetten van de µC, niet tijdens normaal gebruik. Om dat te voorkomen, wordt de Uno R4 gevoed vanuit het shield in plaats van andersom.

Waarom zoveel leds?

Op het OTP Crypto Shield zit een aantal leds. LED1 is een voedingsspanningsindicator, terwijl LED2 en LED3 de activiteit op de RTS- en CTS-lijnen van UART0 van de µC laten zien. LED4 geeft aan dat er data wordt verzonden via de audiopoort van het modem (van UART1), LED5 meldt dat er data wordt ontvangen. LED7 licht op bij een ontvangstfout. LED6 brandt continu tijdens een soft reset (als S1 is ingedrukt en het verzoek wordt verwerkt door de µC), en knippert tijdens het zoeken naar de sleutelpointer als de µC probeert een bericht (of een bestand in Offline-modus) te ontcijferen als de pointerwaarde verloren is gegaan. Als een te lage spanning is gedetecteerd, of na indrukken van S3, knipperen LED5 en LED6 beurtelings om aan te geven dat de statusfile ('.STT', zie onder) succesvol is bijgewerkt. Als ze tegelijk knipperen, is het statusbestand niet geüpdatet (omdat de inhoud niet is veranderd of omdat het schrijven is mislukt).

Audiokanaal

Met een telefoon of walkietalkie en FSK-modem (IC3) kunnen we via een audiokanaal een versleuteld bericht verzenden of ontvangen. IC3 werkt op 1200 baud en maakt gebruik van FSK-frequenties van 1300 Hz en 2100 Hz. Volgens de Wet van Carson ligt de bandbreedte van het analoge signaal tussen 700 en 2700 Hz, zodat we het zelfs over een audiokanaal van lage kwaliteit kunnen versturen. P1 stelt de drempelwaarde in om onderscheid te maken tussen de twee frequenties ('0' of '1'). Om deze af te regelen, zenden we een bericht vanaf een ander shield met een zwak signaal in online-plaintext-modus (optie 5, zie onder). P2 staat in de middenstand voor verbinding met de microfoon of de lijningang van de meeste apparaten. Zet hem in een lage stand bij het instellen van P1 op het andere shield.

Optocoupler IC1 wordt gebruikt om de praatknop te bedienen bij gebruik van een simplex apparaat zoals een walkietalkie.

Eigenlijk kan ieder analoog kanaal (directe draadverbinding, analoge telefoon) of digitaal kanaal met verliesvrije compressie (zoals G.711, A-law of µ-law) worden gebruikt voor de online-modus. Dat geldt ook voor smartphones die zijn verbonden met het internet, als ze werken met een verliesvrije VoIP-codec. Sluit die aan op het OTP Crypto Shield via de headset-aansluiting. Codecs die niet verliesvrij zijn, zoals die worden gebruikt bij GSM, Skype of Viber zijn niet geschikt. Ze zijn prima voor menselijke spraaksignalen, maar niet voor een FSK-signaal, omdat hierbij de toonhoogte veel sneller verandert dan bij een menselijke stem.

Hoe werkt het?

Stel dat twee mensen willen communiceren via OTP Crypto Shields. Dan moeten beide partijen beschikken over hetzelfde OTP-sleutelbestand (KEY, gegenereerd door de True Random Number Generator TRNG). Zoals gezegd mag dezelfde deelreeks van toevallige bytes uit het sleutelbestand nooit meer dan één keer worden gebruikt. Beide partijen moeten dus een soort gesynchroniseerde pointer in het sleutelbestand hebben. Wanneer de synchronisatie verloren is gegaan (om welke reden ook), moet het ook mogelijk zijn om de pointer te herstellen. Om te voorkomen dat dezelfde deelreeks opnieuw wordt gebruikt (wat fataal zou zijn voor de veiligheid van het OTP-systeem), moeten beide partijen beide pointers kunnen beheren.

Daarom moeten beide partijen naast het KEY-bestand ook een statusbestand (STT), met hun eigen pointer (en andere besturingsdata), en een 'mirror'-bestand (MIR) hebben. Dat laatste, dat wordt verzonden naar of ontvangen van de andere partij, bevat de versleutelde versie van

het STT-bestand en wordt, alleen in offline modus, gebruikt voor authenticatie en statuscontrole.

Een voorbeeld

Alice en Bob, de “good guys”, proberen gevoelige informatie uit te wisselen. De kwade Eve probeert de code te kraken door passief af te luisteren, terwijl de boosaardige Mallory gaat proberen de communicatie te verstoren of de inhoud van de uitgezonden berichten of files te veranderen.

De SD-kaart van Alice bevat de files BOB.KEY, BOB.STT en BOB.MIR, en de kaart van Bob bevat ALICE.KEY, ALICE.STT en ALICE.MIR. Beide KEY-files zijn identiek. Ze zijn gegenereerd door de TRNG en bevatten 16 MB tot 4 GB aan toevallige data. De STT-files zijn maar klein (niet meer dan 512 bytes) en bevatten voor mensen leesbare tekst die kan worden bewerkt met een teksteditor zoals Notepad:

```
ALICE.STT:
POINTER :0000000000 [bytes]
TOTAL SIZE :0016244736 [bytes]
CLUSTER :0000007287 [clusters]
CRC32 :2989374417

Alice in Wonderland
Meer informatie...
```

BOB.STT:

```
POINTER :0000000000 [bytes]
TOTAL SIZE :0016244736 [bytes]
CLUSTER :0000007287 [clusters]
CRC32 :2989374417

Sponge Bob Square Pants
Meer informatie...
```

Uit de waarden van POINTER en TOTAL SIZE is af te leiden wanneer een nieuw OTP-sleutelbestand nodig is: de POINTER groeit langzaam naar de waarde van TOTAL SIZE. CLUSTER bevat het nummer van een cluster op de SD-kaart waar POINTER op dit moment naartoe wijst in het sleutelbestand. CRC32 is een checksum die is berekend uit de eerste 16 bytes van de huidige CLUSTER en vier bytes van de waarde van POINTER (POINTER is een 32-bits lang getal dat is opgeslagen in het SRAM van de ATmega328PB in little endian-formaat). Een verandering van die waarden is daardoor te detecteren. Als de waarde van POINTER handmatig wordt veranderd, zal de µC op de SD-kaart gaan zoeken naar de juiste CLUSTER, de CRC32 berekenen en dan het STT-bestand dienovereenkomstig bijwerken. De waarde van CLUSTER op de beide SD-kaarten hoeft niet identiek te zijn voor gelijke waarden van POINTER, omdat de files waarschijnlijk op verschillende clusters staan op verschillende SD-kaarten (en ze kunnen ook gefragmenteerd zijn).

Belangrijk is dat het STT-bestand begint met de string ‘POINTER ’ (let op de spatie). We beschouwen een poging om het versleutelde MIR-bestand te ontcijferen als geslaagd (wat aangeeft dat we de juiste waarde van POINTER hebben gebruikt) als deze string van 8 letters (inclusief de spatie aan het eind) te voorschijn komt bij het ontcijferen van de eerste acht tekens van het MIR-bestand. De vier waarden in het STT-bestand zijn 10-cijferige getallen die meteen na een dubbele punt beginnen. Zo leest de µC ze uit een STT-bestand.

Offline-modus

Als Alice een versleuteld bestand naar Bob wil zenden in offline-modus, gaat ze als volgt te werk:

1. Ze plaatst haar SD-kaart in haar pc en slaat het te coderen bestand erop op. Hier een MP3-bestand met de naam “MUSIC.MP3”.
2. Ze kopieert BOB.STT naar BOB.MIR op haar SD-kaart (ze overschrijft oude inhoud, die is nu niet belangrijk). BOB.MIR bevat nu dus leesbare tekst, identiek aan BOB.STT.
3. Ze steekt de SD-kaart in het OTP Crypto Shield en schakelt het in.
4. Ze bekijkt de inhoud van de root-map (optie 1, **figuur 4**).
5. Ze kiest via de *peer*-file naar wie ze het versleutelde bestand toe wil sturen. Er kunnen meerdere *peer*-files op één SD-kaart staan (voor verschillende contacten in het adresboek: NAAM.KEY, NAAM.STT, NAAM.MIR). Voer alleen de naam van de *peer* in, de extensies worden automatisch toegevoegd (**figuur 5**).
6. Ze kiest optie 3, “Offline file Encryption” en voert de volledige bestandsnaam inclusief extensie in. De µC codeert dan eerst het mirror-bestand BOB.MIR en daarna het hoofdbestand MUSIC.MP3. Als de versleuteling is volbracht, werkt de µC de status in BOB.STT bij (**figuur 6**).
7. Alice verwijdert nu de SD-kaart uit het OTP Crypto Shield en plaatst hem weer in de computer. Ze kan het versleutelde MUSIC.MP3 nu niet meer afspelen. BOB.MIR ziet er nu uit als in **figuur 7** (ook versleuteld). BOB.STT bevat nog steeds leesbare tekst, maar ziet er nu uit als in **figuur 8**. De POINTER is opgeschoven en ook de CLUSTER-waarde is aangepast. De CRC32-waarde is dus ook veranderd. MUSIC.MP3 was circa 5 MB, wat ook blijkt uit de nieuwe POINTER-waarde.
8. Nu stuurt ze een email met de versleutelde bestanden MUSIC.MP3 en BOB.MIR

Figuur 4: Kies optie 1 om de inhoud van de root-map te bekijken.

Figuur 5: Kies 2 om een contact uit het adresboek te selecteren.

Figuur 6: Het coderen van het bestand is volbracht.

met een computer of smartphone of een andere onveilige manier van communicatie. Alice kan de naam van BOB.MIR veranderen in ALICE.MIR, maar dat kan Bob ook doen als hij het bestand ontvangt. Het is beter om beide files een andere naam te geven voor extra veiligheid.

Figuur 7: Het bestand BOB.MIR is nu ook versleuteld.

Offline ontvangen

Bob's SD-kaart bevat de files ALICE.KEY en ALICE.STT (**figuur 9**). Bob's POINTER staat nog op nul en zal 0005380096 bereiken als de ontcijfering klaar is. Als Bob de files van Alice ontvangt, ontcijfert hij ze als volgt:

1. Bob kopieert de twee versleutelde files die hij van Alice heeft ontvangen (MUSIC.MP3 en ALICE.MIR) naar zijn SD-kaart.
2. Bob plaatst zijn SD-kaart in zijn OTP Crypto Shield. De root-map van de SD-kaart ziet er nu uit als in **figuur 10**.
3. Nadat hij de files met de naam van de partner heeft gekozen (optie 2) en er wat initialisatie is uitgevoerd, staat het systeem klaar om te beginnen met decoderen (optie 4, **figuur 11**).

De OTP Crypto Shield probeert ALICE.MIR te ontcijferen met de pointerwaarde uit Bob's ALICE.STT-bestand. Het sleutelbestand wordt doorzocht totdat de juiste beginwaarde voor de pointer is gevonden. Ook nu wordt het decoderen als geslaagd beschouwd als de eerste acht tekens (inclusief de spatie aan het eind) van het MIR-bestand worden gelezen als 'POINTER '. Na succesvol decoderen wordt de inhoud van het MIR-bestand weergegeven. Bob kan nu verschillende zaken controleren:

- De vastgelegde positie van Alice's pointer (0000000000 voordat ze begon met versleutelen) moet overeenkomen met de waarde van Bob's pointer voordat hij begon met decoderen. Als Alice's pointer een lagere waarde heeft dan Bob's pointer (de µC zoekt de juiste positie sowieso), dan is de veiligheid mogelijk in gevaar omdat Alice een OTP-reeks opnieuw heeft gebruikt. Eve zou dan een deel van hun eerdere berichten mogelijkerwijs kunnen decoderen.
- Als er vreemde tekens verschijnen na het decoderen van het MIR-bestand, kan dat erop wijzen dat Mallory heeft geknoeid met het bericht, bijvoorbeeld door de pointerwaarde in het MIR-bestand te veranderen. Een manier om zo'n aanval te voorkomen is door een paar korte, betekenisvolle authenticatieberichten te verstoppen tussen de vier 10-cijferige waarden vóór het coderen van het MIR-bestand (maar zonder het formaat van de vier waarden te veranderen of dubbele punten te gebruiken). Dit verandert de positie van de tekens in het MIR-bestand, waardoor Mallory waarschijnlijk het verkeerde byte verandert.
- Bob kan alle waarden in het ontcijferde MIR-bestand handmatig controleren om onbevoegde pogingen tot verandering van Mallory te detecteren.

Figuur 8: Het bijgewerkte bestand BOB.STT blijft altijd in leesbare tekst.

Figuur 9: Het bestand ALICE.STT op Bob's SD-kaart.

Figuur 10: De root-map van Bob's SD-kaart.

Figuur 11: Het decoderen van het bestand is afgerond. Het bestand moet worden afgespeeld om uit te vinden of alles naar wens is verlopen.

Figuur 12: In het statusbestand op Bob's SD-kaart moet de waarde van de pointer verhoogd.

4. Na het ontcijferen van MUSIC.MP3 kan Bob de SD-kaart uit zijn OTP Crypto Shield halen en in zijn computer steken. Nu kan hij luisteren naar MUSIC.MP3, maar hij kan ook de inhoud controleren van zijn ALICE.STT-bestand (**figuur 12**). Beide pointers zijn gesynchroniseerd en het systeem is klaar om de volgende communicatiesessie te coderen.

Online-modus

In online-modus controleert het OTP Crypto Shield voortdurend de pointerwaarde en waarschuwt als deze afneemt, wat betekent dat een OTP-reeks opnieuw wordt gebruikt. Na het kiezen van een contact (op dezelfde manier als in offline-modus) en initialisatie van de online encryptiemodus, sturen de apparaten aan beide kanten de string "Rd" (ready). Degene die dat ontvangt, mag het eerste bericht gaan versturen.

Elk bericht dat Alice verstuurt bevat:

- de pointer (4 bytes);
- het bericht (payload, plaintext, N bytes);
- Een CRC16-checksum (2 bytes) berekend over de pointer en de payload.

Deze reeks van 4+*N*+2 bytes wordt versleuteld met OTP en er wordt een tweede CRC16-checksum (opnieuw 2 bytes) berekend over de versleutelde reeks en toegevoegd.

Online ontvangst

Als Bob het bericht ontvangt, gaat het decoderen als volgt:

Eerst wordt de checksum van het versleutelde bericht berekend om te detecteren of er communicatiefouten zijn opgetreden. Als de checksums niet overeenkomen, wordt de string "Er" terug naar Alice gestuurd als verzoek om opnieuw te verzenden. Als de checksum correct is, wordt "Ok" verzonden. De string "To" geeft aan dat er een time-out is opgetreden als de bevestiging van de partner te lang op zich heeft laten wachten.

Misschien heeft Mallory enkele bytes in het bericht veranderd en de laatste twee

ONDERDELENLIJST

Weerstanden

(Standaard 0805, 0,125W, 5%)
R1,R18,R24 = 1k8
R2,R16,R17 = 10 k
R3,R21,R22 = 3k3
R4,R10,R11 = 680 Ω
R5,R12,R14,R15 = 1 k
R6,R7,R23 = 4k7
R8 = 22 k
R9 = 7k5, 1%
R13 = 470 Ω
R19 = 100 k
R20 = 1k5
P1 = 10 k instelpotmeter
P2 = 1 k instelpotmeter

Condensatoren

(Standaard SMD 0805)
C1,C2,C4,C5,C6,C7,C8, C9,C13,C16,C17,C18,C20 = 100 n
C3 = 100 µ/10 V, Case B
C10,C12 = 1 µ
C14,C15 = 33 p
C11 = 0,22 F/5,5 V, supercap, steek 5 mm
C19 = 2,2 µ/16V, Case A

Halfgeleiders

IC1 = HCPL-817-000E
IC2 = MAX3232CPWR
IC3 = TCM3105NL
IC4 = LE33CZ-TR
T1 = BC847C
T2 = TSM2302CX RFG
LED1,LED2,LED3 = led, groen, 0805
LED4,LED5,LED6 = led, geel, 0805
LED7 = led, rood, 0805

Diversen

K1 = 8-pens pinheader, steek 2,54 mm
K2 = 10-pens pinheader, steek 2,54 mm
K3,K5 = 6-pens pinheader, steek 2,54 mm
K4 = 12-pens pinheader, steek 2,54 mm
K6 = micro-SD-slot, Molex 47309-serie
K7,K8,K9,JP1 = 2-pens pinheader, steek 2,54 mm
K10 = micro-USB-connector, type B female
S1,S2,S3 = drukknop, 6x6, 2 mm, SMD
X1 = 4,433619 MHz kristal
MOD1 = 12-pens (2x6) pinheader, steek 2,54 mm
DIP-voetje, 16-pens
Print 160510-1, zie 'In de Store'
WIZ107SR Serieel-naar-Ethernet-module
Elektor Uno R4

bytes aangepast om de CRC16 te laten kloppen. Zij kan dat heel gemakkelijk doen. Als ze gewoon probeert het communicatiekanaal te blokkeren, ontstaat er een fout bij het berekenen van de checksum.
Bob's OTP Crypto Shield zal dan proberen het bericht te ontcijferen door te scannen naar de juiste beginwaarde voor de pointer in de huidige cluster van de SD-kaart en de clusters ervoor en erna. Het decoderen wordt als geslaagd beschouwd als de ontcijferde CRC16 klopt en als de ontcijferde pointer overeenkomt met die in Bob's OTP Crypto Shield. Na succesvol decoderen wordt de string "Dc" naar Alice verzonden.
Het is voor Mallory erg moeilijk om ook deze bescherming te omzeilen. Elke vorm van knoeien aan het bericht heeft tot gevolg dat het niet kan worden ontcijferd (terwijl de checksums van de versleutelde berichten wel klopten). Zo zullen zowel Alice als Bob ontdekken dat Mallory probeert de communicatie te verstoren. Als Mallory een eerder opgenomen bericht afspeelt, wordt het wel goed gedecodeerd, maar dan is de pointerwaarde te laag en zal het OTP Crypto Shield van Alice of Bob dat aangeven.

Conclusie

Cryptografie is één van de zeer weinige gebieden waar we nog niet kunnen vertrouwen op kant-en-klare, betrouwbare principes zonder dieper inzicht in het hele proces. De beveiliging van data is een ingewikkeld proces, geen product. Daarom is elk crypto-apparaat nog altijd vatbaar voor verbetering.
Het hier gepresenteerde apparaat is ontworpen om veel mogelijke aanvallen, waarvoor een gewone PC vatbaar is, te voorkomen. De PC is op dit moment het meest gebruikte apparaat voor cryptografie. Ons systeem is ontworpen om veiliger te zijn dan een PC of een black box-cryptofoon en ik geloof dat het dat ook is. Als u veiligheidsaspecten ontdekt die ik over het hoofd gezien heb, plaats uw suggesties dan op de Elektor Labs-pagina van dit project [6]. Ik heb bijvoorbeeld gebruik gemaakt van een CRC, maar een niet-lineaire hashfunctie is waarschijnlijk beter. Nu nog beschermen tegen TEMPEST-aanvallen, maar dat is een ingewikkeld probleem dat afzonderlijk moet worden aangepakt. ⏮

(180543-04)

Tips voor nog meer veiligheid

- Defragmenteren van het sleutelbestand maakt het zoeken van de pointer veel sneller als nodig is om naar een vorige cluster te gaan op de SD-kaart.
- Formatteer de SD-kaart als FAT32 met een clustergrootte van 8-16 kB.
- Begin uw online communicatiesessie met een kort bericht voor een snelle synchronisatie van de pointer. Het duurt langer om de juiste pointerwaarde te vinden bij een lang bericht.
- Als de pointer-synchronisatie verloren is gegaan, is het altijd nuttig om de buffer te 'flushen' met een lange betekenisloze reeks data. De beste lengte is afhankelijk van hoe ver u denkt dat uw pointer is teruggezet. Dit zal u beschermen tegen Eve. Twee betekenisvolle tekstberichten die zijn versleuteld met dezelfde OTP-reeks zijn voor Eve voldoende om de code te kraken.
- Als uw pointer een aanzienlijk lagere waarde heeft dan de andere, verhoog die waarde dan met de hand in het STT-bestand. Het is altijd beter om een deel van de sleutel te verspillen dan een gedeelte tweemaal te gebruiken.
- Om de 'Mallory detectie' te verbeteren, kunt een paar betekenisvolle berichten (zonder dubbele punt) verstoppen in uw MIR bestand voordat u begint met de offline encryptie. Het MIR-bestand zou er ongeveer zo uit kunnen zien

- De Elektor Uno R4-bootloader kan nuttig zijn bij het ontwikkelen en debuggen, maar om veilig te werken moeten we hem uitschakelen via de fuse-instellingen en verwijderen uit het flashgeheugen. De uiteindelijke versie van de firmware flashen we zonder bootloader via de ISP-poort. Een bootloader die bij het opstarten wordt geactiveerd, kan een gedeelte van het flashgeheugen overschrijven en instabiel gedrag en onveilige communicatie veroorzaken.

IN DE STORE

→ 160510-1 OTP Crypto Shield (kale print)
www.elektor.nl/crypto-shield-160510-1

Weblinks

[1] Elektor TV over Cryptografie: www.elektormagazine.com/news/elektor-tv-historic-criptography
[2] Wim van Eck, "Electromagnetic Radiation from Video Display Units: An Eavesdropping Risk?", 1985: https://cryptome.org/emr.pdf
[3] Bruce Schneier, "Secrets & Lies: Digital Security in a Networked World", Wiley, 2004
[4] Wikibooks over OTP: https://en.wikibooks.org/wiki/Cryptography/One_time_pads
[5] Luka Matić, Elektor Magazine maart/april 2017, "Truly Random-Number Generator": www.elektormagazine.com/150116
[6] OTP Crypto Shield bij Elektor Labs: https://www.elektormagazine.com/labs/one-time-pad-otp-crypto-system
[7] Projectpagina: https://www.elektormagazine.com/180543

Halfduplex-telemetrie op 2,4 GHz

goedkoop en eenvoudig met kant-en-klare modules

Reinhardt Weber, DC5ZM / AI6PK (Duitsland)

Veel apparaten kunnen via een (meeraderige) kabel op afstand worden bediend. Maar vaak is het installeren niet zo gemakkelijk en is het resultaat niet echt mooi om te zien. De auteur stond voor de taak om twee dikke, zesaderige besturingskabels te vervangen die van zijn amateurradio-shack naar een rotorbesturing op het dak liepen. Het resultaat was een universeel inzetbaar project dat beschikt over vier digitale en analoge HF-kanalen in halfduplex-modus.

Het project bestaat uit twee 'elektronicablokken', een zender in de shack en een ontvanger bij de rotorbesturing, die via een HF-kanaal met elkaar zijn verbonden. Omdat beide eenheden kunnen zenden en ontvangen (niet tegelijk, maar alleen om de beurt), hebben we te maken met een halfduplex-verbinding. Het is dus eerlijker om niet over zender en ontvanger te spreken maar over front-end (de besturingseenheid) en back-end (de rotorbesturing). De principe-opbouw van het telemetriesysteem is te zien in **figuur 1**.

Modules in front-end en back-end

De hardware van de beide eenheden bestaat voor het grootste deel uit kant-en-klare modules. Aan beide kanten wordt een Arduino Nano gebruikt als controller. De draadloze verbinding wordt tot stand gebracht met aan beide kanten een 2,4-GHz zend/ontvangstmodule van het type nRF24.

Front-end en back-end verschillen alleen in de gebruikersinterface: het back-end (**figuur 2**) heeft analoge ingangen en

gebufferde digitale uitgangen en vier LED's (voor de testfase), en het front-end (**figuur 3**) heeft druktoetsen en een LC-display. Ook daarvoor wordt een kant-en-klare module gebruikt, namelijk een spotgoedkoop *LCD Keypad Shield*.

De Arduino Nano

Over de Arduino Nano [1][2] hoeven we niet veel te vertellen. Hij is functioneel vrijwel compatibel met het alom bekende standaard-instapmodel Arduino Uno, maar dan kleiner en goedkoper. Beide schakelingen worden gevoed met netadapters. De Arduino Nano kan weliswaar gevoed worden met een ongeregelde voedingsspanning van 6...20 V, maar hij wordt dan tamelijk heet als hij een relatief hoge ingangsspanning voor zijn kiezen krijgt — om de simpele reden dat hij intern slechts een simpele lineaire spanningsregelaar heeft. Bij een ingangsspanning van +7...9 V valt het nog mee. Als de spanning van de netadapter hoger is, moet een serieweerstand van 47 Ω in de aansluiting worden opgenomen die (net als de diode als ompoolbeveiliging) een deel van het overtollige vermogen omzet in warmte.

De draadloze module nRF24

Deze populaire radiomodules, die op 2,4 GHz zenden en ontvangen, worden in veel draadloze toepassingen ingebouwd; ze zijn ook al vaak gebruikt in Elektor-projecten. Ze zijn niet alleen goedkoop, maar er zijn in de Arduino-IDE ook bibliotheken beschikbaar die deze modules ondersteunen. De modules werken grotendeels autonoom, we hoeven in het programma alleen maar de bibliotheek mee te linken en dan gaat de draadloze communicatie bijna helemaal vanzelf. De nRF24-modules worden aangestuurd via de SPI-bus van de Arduino Nano. Daarbij zijn maximaal 128 kanalen mogelijk. De chips die met elkaar communiceren, moeten natuurlijk op hetzelfde kanaal werken.

De modules, breakout-boards voor de Nordic-chips nRF24L01, zijn verkrijgbaar in talloze verschillende versies, met printantenne of met een externe SMA-antenne, high- of ultra low-power, met 3,3-V-spanningsregelaar aan boord of zonder, met geïntegreerde vermogenseindtrap en LNA-ontvangstversterker en vooral ook met verschillende pinconfiguraties. Welke versie het meest geschikt is, hangt af van de situatie ter plaatse. Als u alleen maar door een houten dak heen wilt communiceren, dan is de module met printantenne [3] ruim voldoende; als er tussen front- en back-end meerdere verdiepingen met gewapend beton zitten, moeten u teruggrijpen op de versie [4], die de grootste zend- en ontvangstprestaties belooft. Het bereik van 1000 m dat door de fabrikant wordt gespecificeerd, is alleen haalbaar in de open lucht onder optimale omstandigheden. Onder de gegeven omstandigheden zou een bereik dat is te vergelijken met

Figuur 1: Blokschema van het telemetriesysteem met front-end en back-end.

Figuur 2: Schema van het back-end met Arduino en draadloze module.

Figuur 3: Het front-end bestaat uit een Arduino, een draadloze module en een *LCD Keypad Shield*.

Figuur 4: De communicatie tussen de beide eenheden in de vorm van een stroomschema.

dat van een WLAN-router realistisch zijn. Trouwens: Nordic zelf raadt de nRF24L01 niet meer aan voor nieuwe ontwikkelingen, maar verwijst naar het nieuwe, krachtigere model nRF24L01+. Maar de voorraden van 'oude' breakout-boards zullen nog wel enkele jaren groot genoeg zijn om aan de vraag te voldoen.

De zend/ontvangstmodule werkt niet, zoals de Arduino, met 5 V maar met 3,3 V. In principe zouden we de transceivermodule rechtstreeks kunnen voeden uit de gestabiliseerde 3,3-V-uitgang van de Arduino Nano. Die kan maximaal 50 mA leveren. Voor de eenvoudige module met printantenne, die zo'n 11...12 mA gebruikt, zou dat ruim voldoende zijn, maar niet voor uitvoering met versterkers, omdat die bij zenden maximaal 115 mA trekt. Daarom wordt de draadloze module hier gevoed met een extra spanningsregelaar die wordt aangesloten op de 5-V-aansluiting van de Arduino. Er bestaat voor dit doel overigens een speciale module, maar wij maken, heel klassiek, gebruik van een spanningsregelaar (SMD of TO92) samen met de gebruikelijke ontkoppelcondensatoren.

LCD Keypad Shield

Het '1602 LCD Keypad Shield For Arduino', dat onder andere verkrijgbaar is bij de firma DFRobot uit Shanghai [5], past mechanisch op de Arduino Uno, maar kan natuurlijk ook probleemloos met de Nano worden gebruikt. Het display met 2x16 karakters wordt niet serieel aangestuurd via SPI of I²C, maar via de parallelle bus. De displaypennen DB4...DB7 zijn verbonden met D4...D7 op de header, de reset-besturingspen zit op D8 en enable op D9. Het display is uitgerust met een instelpotmeter voor het instellen van het contrast.

De vijf druktoetsen SELECT, LEFT, UP, DOWN en RIGHT zijn op een wat ongebruikelijke manier aangesloten. Ze zijn niet gewoon elk apart verbonden met een digitale aansluiting, maar zijn samen verbonden met de analoge aansluiting A0 (zie t**ekstkader 1**). De druktoetsen zijn intern verbonden met een spanningsdeler. De analoge spanning op A0 van de controller moet daarom worden geëvalueerd om te bepalen of er een toets is ingedrukt en zo ja, welke dat dan is. Dat betekent ook, dat er altijd maar één toets tegelijk mag worden ingedrukt. Het shield heeft nog een zesde druktoets (RESET), die met de RST-pen van de Arduino is verbonden, maar die wordt in deze applicatie niet gebruikt. Thomas Clausen heeft dit shield nader onderzocht en beschrijft het onder [6].

Het verloop van de halfduplex-communicatie

Het verloop van de communicatie tussen front-end en back-end is weergegeven in de flowcharts (**figuur 4**). In principe registreert het front-end alleen het indrukken van een knop en die informatie wordt naar het back-end gestuurd, waar dan de verwerking plaatsvindt. De toetsen UP, DOWN, LEFT en RIGHT maken dat één van de digitale uitgangen omlaag wordt getrokken. Met de toets SELECT kan worden omgeschakeld tussen vier mogelijke analoge ingangen om één van de vier meetwaarden in te lezen.

In het programma voor het front-end (TX.ino) bewaakt de controller daarom in de hoofdlus, na de initialisatie, de status van de toetsen door de analoge spanning op A0 te meten. Hij bepaalt aan de

Figuur 5: De LCD-weergave voor de rotorbesturing.

Figuur 6: Voorbeelden van de LCD-weergave voor verschillende toepassingen.

hand daarvan welke van de vijf toetsen is ingedrukt. Let op: er zijn verschillende varianten van het shield in de handel en die kunnen verschillende waarden van de weerstanden in de spanningsdeler hebben, zodat het nodig kan zijn om de drempelwaarden in **listing 1** te veranderen om de toetsen betrouwbaar te herkennen. Daarna gaat het programma naar de zendmodus, zendt de informatie over de toets die is ingedrukt naar het back-end, en gaat dan over naar ontvangstmodus.

Het back-end, dat na de start van het programma RX.ino in ontvangstmodus staat, bestuurt aan de hand van de ontvangen informatie één van de vier digitale uitgangen of schakelt om naar een andere analoge ingang. Daarna leest het programma de status van de vier digitale uitgangen, het ingestelde analoge kanaal en de analoge spanning op dat kanaal in. De gemeten spanning wordt omgezet in een 10-bit digitale waarde. Het back-end schakelt om naar zendmodus en stuurt al deze data (in gecodeerde vorm) in één keer naar het front-end. Na ontvangst decodeert het front-end de data en geeft de verkregen informatie weer op het LC-display (**figuur 5**).

Bij deze bidirectionele communicatie controleert de zend/ontvang-chip nRF24L01 continu de correcte overdracht. Als er een fout optreedt, wordt een corresponderende melding weergegeven op het display.

Softwarevarianten

De beide programma's, die u van de projectpagina [7] kunt downloaden, bevatten voorbeelden voor verschillende toepassingen. Met de toetsen UP, DOWN, LEFT en RIGHT wordt in de oorspronkelijke toepassing de antennerotor aangestuurd door verschillende uitgangen laag te maken. Maar in de listing van het front-end worden als voorbeeld drie verschillende schakelfuncties getoond: LEFT als momentcontact, RIGHT als aan/uit-schakelaar en UP en DOWN als radiobuttons (schakelaars die elkaar wederzijds beïnvloeden).

De SELECT-toets schakelt om tussen de vier mogelijke analoge ingangen. Het LCD geeft zowel het nummer van de analoge ingang als de gemeten waarde weer. In de listing van het front-end zijn voorbeelden te zien hoe het display eruit zou kunnen zien, als het azimut of elevatie weergeeft (meer analoge kanalen heeft een antennerotor niet nodig). In de programmacode is ook te zien hoe een klein display voor een voltmeter of een thermometer is te realiseren (**figuur 6**). Voor het weergeven van speciale tekens heeft de auteur gebruik gemaakt van de praktische *Character Generator* [8].

Beide programma's zijn bewust eenvoudig gehouden en van commentaar voorzien. Er zijn veel mogelijkheden om ze aan uw eigen behoeften aan te passen. Alle regels en functies die te maken hebben met het aansturen van de nRF24-module, kunt u het beste onver-

Listing 1: Evaluatie van de druktoetssignalen.

```
void get_key_pushed()
{
   message_to_RX = analogRead(A0)/10;                    // values from 0 to 102
   delay(100);    // key debounce

   if (message_to_RX<5){while(analogRead(A0)<5); message_to_RX = 2;} // Right
   if (message_to_RX> 5  && message_to_RX<20 ) message_to_RX = 3;       // Up
   if (message_to_RX>20  && message_to_RX<35 ) message_to_RX = 4;     // Down
   if (message_to_RX>35  && message_to_RX<55 ) message_to_RX = 1;     // Left
   if (message_to_RX>55  && message_to_RX<85 ) message_to_RX = 5;   // Select
}
```

Rotorbesturing

Pen 6 levert 2...4,5 VDC overeenkomend met 0...450° (azimut)
Pen 1 levert 2...4,5 VDC overeenkomend met 0...180° (elevatie)
Pen 4 draai naar links (actief laag)
Pen 2 draai naar rechts (actief laag)
Pen 5 draai omlaag (actief laag)
Pen 3 draai omhoog (actief laag)
Pen 7 nc
Pen 8 gemeenschappelijke massa

Het rotorbesturingsapparaat YEASU G-5600 heeft vier toetsen voor de besturing van de rotor: UP en DOWN voor elevatie en LEFT en RIGHT voor azimut. Als de toetsen worden ingedrukt, trekken ze de overeenkomstige besturingslijn naar massa. Deze besturingslijnen zijn, naast de drukknoppen, ook beschikbaar op een 8-polige DIN-bus aan de achterkant van het apparaat.
Verder geeft de rotorbesturing het instellen van het azimut en de elevatie weer op twee grote wijzerinstrumenten. De beide analoge spanningen voor deze meetinstrumenten komen van de rotor zelf. Ze liggen in het bereik 0...4,5 V en geven de elevatie van 0...180° en de azimut van 0...450°. Ook die twee analoge spanningen zijn te vinden op de DIN-bus aan de achterkant van het apparaat, zodat een complete afstandsbediening mogelijk is.
Er is dus niets op tegen om het besturingsapparaat weg te laten, zodat alleen korte besturingskabels voor de rotor nodig zijn. We sluiten een kleine besturingseenheid aan, die alleen via een draadloze ISM-verbinding verbonden is met shack.

anderd laten, tenzij u voldoende ervaring heeft met die module.

Realisatie

De schema's en layouts zijn gemaakt met het programma EAGLE en zijn te vinden op de projectpagina [7]. De printlayouts hebben tamelijk brede koperbanen, en de printen zijn dan ook geschikt om zelf te etsen. Als de hardware is opgebouwd, kunt u de beide besturingsprogramma's in de Arduino's laden met de Arduino-IDE. De programma's maken gebruik van de bibliotheken LiquidCrystal.h, SPI.h, nRF24L01.h en RF24.h. Sommige van die bibliotheken zijn bij uw Arduino-IDE misschien al geïnstalleerd. Als dat niet het geval is, zijn ze te vinden onder [9][10][11][12]. U installeert ze in de IDE via *Schets → Bibliotheek gebruiken.*

Omdat consequent gebruik is gemaakt van modules, kunt u de zender en de ontvanger elk op een halve enkelzijdige eurokaart opbouwen, als u bereid bent om een paar draadbruggen aan te brengen. De vijf gebruikte modules zijn op internet te koop voor in totaal nog geen € 30, maar er zijn allerlei versies verkrijgbaar. Over het algemeen zijn Chinese aanbieders niet zo scheutig met documentatie, zodat het vaak nog een hele klus is om uit te vinden wat u nou eigenlijk gekocht hebt. Daarom is het verstandiger in zee te gaan met leveranciers die schema's, layouts en programmavoorbeelden beschikbaar stellen. ⏮

(180280-04)

Dit project is ook gepubliceerd in het tijdschrift CQ DL van de Deutsche Amateur-Radio-Club (editie 4/2018).

IN DE STORE

→ **Draadloze module**
www.elektor.nl/nrf24l01-150499-91

→ **Arduino Nano**
www.elektor.nl/arduino-nano-3

→ **JOY-iT Nano**
www.elektor.nl/joy-it-nano-v3

Weblinks

[1] Arduino Nano: www.elektor.nl/arduino-nano-3
[2] JOY-iT Nano: www.elektor.nl/joy-it-nano-v3
[3] Radiomodule nRF24L01: www.elektor.nl/nrf24l01-2-4-ghz-wireless-module-8-pin-150499-91
[4] nRF2401 met versterker en SMA: www.elecfreaks.com/wiki/index.php?title=2.4G_Wireless_nRF24L01p_with_PA_and_LNA
[5] 1602 LCD Keypad Shield: www.dfrobot.com/product-51.html
[6] LCD Keypad Shield in detail: www.thomasclausen.net/en/walking-through-the-1602-lcd-keypad-shield-for-arduino/
[7] Projectpagina: www.elektormagazine.nl/180280-04
[8] LCD-karaktergenerator: https://omerk.github.io/lcdchargen/
[9] LCD-bibliotheek: https://playground.arduino.cc/Main/LiquidCrystal
[10] SPI-bibliotheek: https://github.com/PaulStoffregen/SPI
[11] RF24-bibliotheek (1): www.arduinolibraries.info/libraries/rf24
[12] RF24-bibliotheek (2): https://github.com/maniacbug/RF24

Elektor Labs Pipeline

Hier zijn weer vier samenvattingen van interessante projectjes die mensen hebben gepost op Elektor Labs dot com. Zo kunt u zien wat onze lezers zoal bezighoudt.

Van simpel tot ingewikkeld, van meten tot regelen — op Elektor Labs kunt u het vinden, en nog veel meer. Werp eens een blik op deze projecten, er zit vast iets bij dat u kunt gebruiken!

Hoe steil is die helling?
Bent u fan van wielrennen en volgt u de grote evenementen, zoals — in chronologische volgorde — de Giro d'Italia, de Tour de France en de Vuelta in Spanje? Dan weet u dat heuvels en bergen 'cols' heten en dat die qua zwaarte in een bepaalde categorie vallen. Maar wat is de categorie van de col die u zelf beklimt? Hoe zwaar hebt u het, in vergelijking met de profrenners? Met dit projectje komt u het aan de weet.

@ Elektor Labs: https://goo.gl/jjcw8x

Stuur uw vertrouwelijke berichten in een veilige envelop
Om te voorkomen dat berichten worden onderschept, of afgetapt, of gewijzigd, kunt u ze versleutelen. Voor de decodering is er dan aan de ontvangstkant wel een decodeersleutel nodig, wat de vraag opwerpt hoe u die ongezien en veilig op de plaats van bestemming krijgt. Dit project lost dit kip-ei-probleem op door het bericht te verpakken in een envelop waaraan te zien is of ermee gerommeld is.

@ Elektor Labs: https://goo.gl/rnk28d

Met drie onderdelen een sinus-oscillator van €/£/$ 1-
Sinusgolven zijn belangrijk in de elektronica. Voor allerlei toepassingen zijn sinussignalen nodig en heel veel elektronica wordt getest door er een sinus op te zetten. Het is dus handig om altijd een sinus-oscillator onder handbereik te hebben op de werkbank. Hebt u die nog niet, maak dan dit project. U hebt er maar drie onderdelen voor nodig en voor de kosten hoeft u het niet te laten.

@ Elektor Labs: https://goo.gl/YRmP1J

Programmeerbare koelkast-waakhond
Met dit projectje hebt u een schakeling die de huidige temperatuur van uw koelkast weergeeft en bewaakt en die bijhoudt wanneer en hoe lang de deur open stond. Bovendien kan ze een alarm triggeren als de temperatuur te hoog wordt en als de deur te lang open blijft staan.

(180571-02)

@ Elektor Labs: https://goo.gl/pBeCuH

Schmitt-triggers met opamps

schakelingen en berekeningen

Een Schmitt-trigger is een van de basisschakelingen van de elektronica. Niets bijzonders zult u denken, je neemt een opamp en een paar weerstanden en dat is het wel zo'n beetje. Als u de hysteresis en de schakeldrempels slechts ruwweg wilt vastleggen, dan hebt u gelijk. Als u deze waarden echter precies wilt hebben en weerstanden uit de standaard-E-reeksen wilt gebruiken, dan zijn aangepaste schakelingen en dito berekeningen nodig. Deze vindt u in dit artikel.

Volker Schmidt (Duitsland)

Het berekenen van Schmitt-triggers met opamps is best lastig als u daar weinig ervaring mee hebt. De berekening begint meestal met het op basis van de terugkoppelweerstand bepalen van de referentiespanning. Die referentiespanning ligt dan meestal niet in het midden van het hysteresisvenster, en ook levert de berekening vaak 'kromme' waarden op waarvoor bijvoorbeeld geen zenerdiodes beschikbaar zijn. Dan blijft voor het leveren van de referentiespanning alleen een spanningsdeler met weerstandswaarden buiten de gebruikelijke E-reeksen over, of een instelpotmeter. Het zou handiger zijn als de referentiespanning met zenerdiodes, referentiespanningsbronnen of eenvoudige spanningsdelers met weerstanden uit de E12-reeks zou kunnen worden opgewekt, en de schakeling aan de hand van die referentiespanning zou kunnen worden berekend. Met de standaardschakelingen is dit niet zonder meer mogelijk, maar na een kleine modificatie kunnen eerst de referentiespanning en de schakeldrempels worden bepaald en vervolgens de andere onderdelen worden berekend. Voor deze modificatie zijn maar drie extra onderdelen nodig, en de methode kan zowel voor niet-inverterende als voor inverterende Schmitt-triggers worden gebruikt.

Niet-inverterende Schmitt-trigger

Het schakelgedrag van de niet-inverterende Schmitt-trigger uit **figuur 1** wordt bepaald door de spanning op R2 en door zijn uitgangsspanningsbereik. We kunnen de schakeldrempels van de Schmitt-trigger bepalen door de hysteresis ΔU_H bij U_{R2} op te tellen en ervan af te trekken. In het ideale geval ligt de

Figuur 1. Standaardschakeling van een niet-inverterende Schmitt-trigger met opamp.

Figuur 2. Hysteresisdiagram van een ideale Schmitt-trigger.

referentiespanning midden tussen de twee schakeldrempels U_{H+} en U_{H-}. Dit is niet echt noodzakelijk, maar u moet er wel op letten dat de referentiespanning niet te dicht bij een van de schakeldrempels komt te liggen. Berekenen we de standaardschakeling van figuur 1 voor de referentiespanning U_{R2}, dan hebben we het probleem dat één van de schakeldrempels niet vrij kan worden gekozen. De waarden van U_{H+} en U_{H-} zijn namelijk niet symmetrisch ten opzichte van de referentie. In theorie zijn ze alleen even groot en symmetrisch als we de inverterende ingang bij een symmetrische voeding aan massa leggen, of bij een asymmetrische voeding precies op de helft van het uitgangsspanningsbereik van de opamp.
Dit wordt veroorzaakt door de stroom die door R4 en R1 loopt. Deze hangt af van het spanningsverschil tussen de uitgangsspanning en de spanning op de inverterende ingang. Alleen als de absolute waarde van deze stroom zowel in ingeschakelde als uitgeschakelde toestand hetzelfde is, dan gelden de volgende vergelijkingen:

$$U_{H+} = U_{R2} + \Delta U_{H+}$$

$$U_{H-} = U_{R2} - \Delta U_{H-}$$

$$|\Delta U_{H+}| = |\Delta U_{H-}|$$

$$I_{R4} = \frac{U_{maxop} - U_{R2}}{R4} = -\frac{U_{minop} - U_{R2}}{R4}$$

$$I_{R1} = -I_{R4}$$

$$\Delta U_H = -I_{R4} \times R1$$

Is dit niet het geval, dan kan bijvoorbeeld U_{H+} wel vooraf worden bepaald, maar hebben we op U_{H-} geen directe invloed. De waarde van I_{R4} is dan in de beide schakeltoestanden niet gelijk. Een rekenvoorbeeld voor figuur 1 met een opamp CA3140 maakt dit duidelijk. We gaan uit van de volgende waarden:

$U_m = U_{R2}$ = 1,7 V	U_B = 9 V	ΔU_{H+} = 0,2 V
U_{H+} = 1,9 V	U_{maxop} = 6,8 V (gemeten)	
U_{H-} = 1,5 V gewenst	U_{minop} = 0,01 V (gemeten)	

Nu berekenen we **spanningsdeler** R2/R3:

$$\frac{R2}{R3} = \frac{U_{R2}}{U_{R3}}$$

$$R2 = R3 \times \frac{U_{R2}}{U_{R3}}$$

Kiezen we voor R3 de waarde 10 kΩ, dan volgt hieruit:

$$R2 = 10\,k\Omega \times \frac{1{,}7V}{7{,}3V} = 2{,}33\,k\Omega$$

Deze waarde kan met 2,2 kΩ + 100 Ω voldoende nauwkeurig worden samengesteld. Omdat tijdens het schakelen de spanningen op beide opamp-ingangen gelijk zijn, geldt ook:

Figuur 3. Standaardschakeling (met berekende waarden) van de niet-inverterende Schmitt-trigger.

$$U_p = U_n$$

$$U_{R4} = U_{minop} - U_{R2}$$

$$U_{R4} = 0{,}01V - 1{,}7V = -1{,}69V$$

Bij een waarde van 10 kΩ voor R4 geldt:

$$I_{R4} = \frac{U_{R4}}{R4} = \frac{-1{,}69V}{10\,k\Omega} = -169\mu A$$

und:

$$I_{R1} = -I_{R4}$$

R1 vinden we nu met de formule:

$$R1 = \frac{\Delta U_{H+}}{I_{R1}} = \frac{0{,}2V}{169\mu A} = 1{,}183\,k\Omega$$

Een waarde van 1,2 kΩ is hier een goede benadering voor R1. Nu gaan we de **onderste schakeldrempel** U_{H-} berekenen:

$$U_{R4} = U_{maxop} - U_{R2}$$

$$U_{R4} = 6{,}8V - 1{,}7V = 5{,}1V$$

$$I_{R4} = \frac{U_{R4}}{R4} = \frac{5{,}1V}{10\,k\Omega} = 510\,\mu A$$

$$R1 = \frac{\Delta U_{H-}}{I_{R1}}$$

$$\Delta U_{H-} = R1 \times I_{R1}$$

$$I_{R1} = -I_{R4}$$

$$\Delta U_{H-} = 1{,}2\,k\Omega \times -510\,\mu A = -0{,}612V$$

$$U_{H-} = U_{R2} - \Delta U_{H-} = 1{,}088V$$

De gevonden onderste schakeldrempel wijkt duidelijk af van de gewenste waarde. In **figuur 3** ziet u de standaardschakeling met de berekende waarden. Uit de berekeningen en de twee

Figuur 4. Oscillogram van de bovenste schakeldrempel van de (standaard) niet-inverterende Schmitt-trigger.

Figuur 5. Oscillogram van de onderste schakeldrempel van de (standaard) niet-inverterende Schmitt-trigger.

oscillogrammen (**figuur 4** en **figuur 5**) blijkt dat de schakeldrempels zowel in theorie als in de praktijk niet symmetrisch liggen ten opzichte van de referentie.

Schakelen met precisie

Aangepaste schakeling

De oplossing is om een gescheiden terugkoppelnetwerk voor de bovenste en de onderste schakeldrempel te gebruiken. Door geschikte waarden voor de weerstanden te nemen kunnen dan beide drempels vrij worden gekozen. Hiervoor moeten we, zoals in **figuur 6** is te zien, naast twee terugkoppelweerstanden ook nog twee diodes toevoegen. Vanwege de geringe spanningsval hebben schottkydiodes zoals de 1N5817 de voorkeur. R4 zorgt voor de bovenste schakeldrempel en R5 voor de onderste. De hysteresis kan hiermee flexibel worden aangepast.

Voor de berekening van deze schakeling (eveneens met CA3140) gelden de volgende uitgangswaarden:

$U_m = U_{R2} = 1{,}7$ V	$U_B = 9$ V	$U_{FD} = 0{,}21$ V (gemeten)
$U_{H+} = 1{,}9$ V	$U_{maxop} = 6{,}8$ V (gemeten)	$R_3 = 10$ kΩ
$U_{H-} = 1{,}5$ V	$U_{minop} = 0{,}01$ V (gemeten)	$\Delta U_H = 0{,}2$ V

Spanningsdeler R2/R3 wordt als volgt berekend:

$$R2 = R3 \times \frac{U_{R2}}{U_{R3}} = 10\,k\Omega \times \frac{1{,}7V}{7{,}3V} = 2{,}33\,k\Omega$$

R2 kan weer uit een 2,2-kΩ- en een 100-Ω-weerstand worden samengesteld. De bovenste schakeldrempel wordt nu:

$$U_{R4} = U_{minop} + U_{FD} - U_{R2} = 0{,}01V + 0{,}21V - 1{,}7V = -1{,}48V$$

Met R4 = 10 kΩ geeft dit een stroom I_{R4} van -148 µA.

Figuur 6. Schakeling van de gemodificeerde niet-inverterende Schmitt-trigger.

Figuur 7. Schakeling (met berekende waarden) van de gemodificeerde niet-inverterende Schmitt-trigger.

Figuur 8. Oscillogram van de bovenste schakeldrempel van de gemodificeerde niet-inverterende Schmitt-trigger.

Figuur 9. Oscillogram van de onderste schakeldrempel van de gemodificeerde niet-inverterende Schmitt-trigger.

Omdat bij de schakeldrempels de spanningen aan de opamp-ingangen gelijk zijn, geldt:

$$U_p = U_n$$

En dus ook:

$$I_{R1} = -I_{R4}$$

De spanning op de niet-inverterende ingang moet

$$\Delta U_{H+} = U_{H+} - U_{R2}$$

hoger zijn zodat de Schmitt-trigger omschakelt en de uitgang naar U_{maxop} springt. De waarde van R1 volgt nu uit de formule:

$$R1 = \frac{\Delta U_{H+}}{I_{R1}} = \frac{0{,}2V}{148\,\mu A} = 1{,}351\,k\Omega$$

De E24-waarde 1,3 kΩ geeft voor R1 slechts een kleine afwijking.
Voor de onderste schakeldrempel geldt:

$$I_{R5} = \frac{U_{maxop} - U_{FD} - U_{R2}}{R5} = 148\,\mu A \rightarrow bovenste\ schakeldrempel!$$

$$R5 = \frac{U_{maxop} - U_{FD} - U_{R2}}{I_{R5}} = \frac{6{,}8V - 0{,}21V - 1{,}7V}{148\,\mu A} = 33{,}04\,k\Omega$$

Voor R5 is de E12-waarde 33 kΩ prima geschikt. In **figuur 7** ziet u de schakeling met berekende waarden. Uit de twee oscillogrammen (**figuur 8** en **figuur 9**) blijkt dat deze goed overeenkomen met de theoretische waarden.

Inverterende Schmitt-trigger

Ook bij de inverterende Schmitt-trigger kunnen de schakeldrempels op basis van de referentiespanning worden berekend. Net als bij de niet-inverterende versie is ook bij de standaardschakeling van **figuur 10** een kleine modificatie mogelijk (zie **figuur 11**). Ook hier worden de schakeldrempels met behulp van twee weerstanden en twee schottkydiodes ontkoppeld. Deze bevinden zich nu niet in het terugkoppelnetwerk, maar in de verbinding met de referentiespanning. De opsplitsing van R2 in R2.A en R2.B zorgt ervoor dat voor beide schakeltoestan-

Figuur 10. Standaardschakeling van een inverterende Schmitt-trigger.

Figuur 11. Schakeling van de gemodificeerde inverterende Schmitt-trigger.

Figuur 12. Schakeling (met berekende waarden) van de gemodificeerde inverterende Schmitt-trigger.

den de juiste vergelijkingsspanning op de niet-inverterende opamp-ingang aanwezig is. Weerstand R3 wordt gekozen, en daaruit volgen de waarden van R2.A en R2.B.
Voor de onderste schakeldrempel geldt: $U_a = U_{minop}$ en $U_H = U_{H-}$

$$\frac{U_{H-} - U_{minop}}{R3} = \frac{U_{R2} - U_{FD} - U_{H-}}{R2.A}$$

$$R2.A = R3 \times \frac{U_{R2} - U_{FD} - U_{H-}}{U_{H-} - U_{minop}}$$

Voor de bovenste schakeldrempel geldt: $U_a = U_{maxop}$ en $U_H = U_{H+}$

$$\frac{U_{maxop} - U_{H+}}{R3} = \frac{U_{H+} - U_{R2} - U_{FD}}{R2.B}$$

$$R2.B = R3 \times \frac{U_{H+} - U_{R2} - U_{FD}}{U_{maxop} - U_{H+}}$$

De gemodificeerde schakeling wordt nu berekend en gecontroleerd op correcte werking. We gebruiken een TL071 waarbij de volgende uitgangswaarden worden gehanteerd:

$U_m = U_{R2} = 5{,}6$ V	$U_B = 9$ V	$R_3 = 10$ kΩ
$U_{H+} = 6{,}6$ V	$U_{maxop} = 8{,}2$ V (gemeten)	$U_{FD} = 0{,}21$ V (gemeten)
$U_{H-} = 4{,}6$ V	$U_{minop} = 1{,}48$ V (gemeten)	

Voor de onderste schakeldrempel geldt de volgende vergelijking:

$$R2.A = R3 \times \frac{U_{R2} - U_{FD} - U_{H-}}{U_{H-} - U_{minop}} =$$

$$\frac{5{,}6V - 0{,}21V - 4{,}6V}{4{,}6V - 1{,}48V} \times 10\,k\Omega = 2{,}532\,k\Omega$$

R2.A kan eenvoudig uit 1,5 kΩ en 1 kΩ worden samengesteld. Voor de bovenste schakeldrempel geldt:

$$R2.B = R3 \times \frac{U_{H+} - U_{R2} - U_{FD}}{U_{maxop} - U_{H+}} =$$

$$\frac{6{,}6V - 5{,}6V - 0{,}21V}{8{,}2V - 5{,}6V} \times 10\,k\Omega = 3{,}038\,k\Omega$$

Als R2.B uit 1,8 kΩ en 1,2 kΩ wordt samengesteld, dan geeft dit een fout van niet meer dan ongeveer 1,3%. In **figuur 12** ziet u de schakeling met de berekende waarden. De oscillogrammen (**figuur 13** en **figuur 14**) laten zien dat de echte drempels dicht bij de berekende waarden liggen.

Conclusie

Hoewel dit nogal droge kost lijkt, zult u bij uw volgende berekening van een Schmitt-trigger merken dat de schakelingen van figuur 7 en figuur 12 met de bijbehorende formules snel

Figuur 13. Oscillogram van de bovenste schakeldrempel van de gemodificeerde inverterende Schmitt-trigger. Het verloop van de uitgangsspanning (de blauwgroene lijn) is ook geïnverteerd.

Figuur 14. Oscillogram van de onderste schakeldrempel van de gemodificeerde inverterende Schmitt-trigger.

Betekenis van de symbolen

Symbool	Betekenis
U_b	Voedingsspanning
U_{maxop}	Maximale uitgangsspanning opamp
U_{minop}	Minimale uitgangsspanning opamp
U_H	Schakelspanning algemeen
U_{H+}	Bovenste schakeldrempel
U_{H-}	Onderste schakeldrempel
U_M	Middelste schakelspanning
ΔU_{H+}	Positieve hysteresis
ΔU_{H-}	Negatieve hysteresis
U_{R2}	Referentiespanning
U_{FD}	Doorlaatspanning Schottky-diode

De waarden van U_{minop}, U_{maxop} en U_{FD} moeten door meting worden bepaald, omdat deze essentieel zijn voor de nauwkeurigheid. De in datasheets van opamps en diodes vermelde waarden hebben vaak een te grote tolerantie.

kunnen worden berekend. Dit loont echter alleen als het op nauwkeurige schakeldrempels aankomt. Is dit niet het geval, dan hebt u genoeg aan de 'normale' schakeling van figuur 3. En ook daarvoor vindt u hier de formules. Veel plezier ermee!

(160340)

Over de auteur

Volker Schmidt heeft een groot aantal jaren als projectingenieur voor op afstand bediende systemen en programmeerbare besturingen gewerkt. Nu is hij als zelfstandige actief in de IT-branche. Elektronica is al sinds zijn twaalfde levensjaar zijn hobby. Tegenwoordig houdt hij zich voornamelijk bezig met AVR-controllers, en daarnaast ook met analoge schakelingen.

Advertentie

Voorkomen is beter dan genezen!

Ruim 42% van de binnenbranden ontstaat door defecte of te warm wordende apparatuur (2013, bron: CBS). Elektor geeft je als cadeautje deze warmtegevoelige sticker die je helpt met het detecteren en voorkomen van mogelijk brandgevaar bij je adapters en apparatuur. Hij verkleurt afhankelijk van de temperatuur en geeft zo op een duidelijke manier aan of het apparaat te warm is, zodat je er alert op kunt reageren en escalatie (brand!) vóór kunt zijn.

Dus... plak hem op je telefoonlader, laptopvoeding of televisieontvanger en voorkom een hoop problemen en gevaren!

Voor Elektor-abonnees is hij gratis bij dit magazine meegeleverd. Maar je kunt hem ook in onze webshop bestellen als je geen lid bent (of als je lid bent en er meer wilt): www.elektor.nl/brand

nederlandse brandwonden stichting

Steun ons via **www.elektormagazine.nl/brandwondenstichting**

PobDuino

Flowcode en Arduino geven oude robots een nieuw leven

Jean-Noël Lefebvre en **Stéphane Huet** (Frankrijk)

Bent u op zoek naar een programmeerbare modulaire robot voor onderwijsdoeleinden? Bouw dan één of meer robots met het PobDuino-board. Dit geeft uw robots bijzondere eigenschappen: ze zijn programmeerbaar in Flowcode (de basislicentie volstaat) en kunnen voor het aansluiten van sensoren en actuatoren gebruikmaken van Arduino-compatibele boards.

Hoe het begon

Stéphane Huet is leraar ingenieurswetenschappen aan het Louis Aragon Lyceum in Givors (Frankrijk). Hij maakt voor zijn robotica-lessen al enkele jaren gebruik van de *Pob*-robots van het bedrijf Pob Technology. Omdat het lyceum actief heeft bijgedragen aan het mechanisch ontwerp van deze educatieve robot, beschikt het nu, zoals veel andere scholen in Frankrijk, over een 'robotpark' van zo'n tien exemplaren.

De belangrijkste voordelen van dit robotplatform zijn:

- mechanische robuustheid en modulariteit;
- veel verschillende accessoires (servomotoren en sensoren);
- de mogelijkheid om niet alleen in C te programmeren, maar ook met de grafische taal Orbee (flowcharts), die door Pob Technology werd ontwikkeld.

Helaas bestaat Pob Technology niet meer, en is de *Pob* nu een weeskind.

Stéphane heeft geprobeerd om zijn *pob*-robots een tweede leven te geven. Op zijn varlanglijstje hiervoor staat onder andere:

- de mechatronische onderdelen (mechanisme, accu's, motoren, sensoren en actuatoren) moeten behouden blijven;
- de robot moet met flowcharts kunnen worden geprogrammeerd (zoals bij Orbee);
- het platform moet toegang krijgen tot de Arduino-omgeving, zodat kan worden geprofiteerd van het overvloedige aanbod van hardware-modules en open-source software-bibliotheken.

Figuur 1. De verschillende functieblokken van het PobDuino-systeem met 'dubbele besturing'.

Dubbele besturing

Wat betreft de software is om twee redenen niet gekozen voor Scratch.
Ten eerste is Scratch niet gebaseerd op flowcharts maar op een grafische weergave met blokstructuren die niet past bij het onderwijskundige doel van het project. In de tweede plaats kan Scratch niet zelfstandig op een Arduino worden uitgevoerd, er is een draadloze verbinding nodig met een computer waar Scratch op draait (of een embedded Raspberry Pi met Scratch). Na wat onderzoek kwam Jean-Noël bij Flowcode terecht, een serieuze kandidaat die bij veel scholen wordt gebruikt. Hierbij moesten echter eerst nog wat problemen worden opgelost:

- De uitgever van Flowcode heeft het over compatibiliteit met Arduino, maar dan bent u aangewezen op de hardware van specifieke Arduino-boards. Bovendien is er geen softwarematige integratie: het is niet mogelijk om direct bibliotheken uit de Arduino-IDE te importeren :-(
- Met de basislicentie van Flowcode voor Arduino/AVR zijn de mogelijkheden erg beperkt. Om bijvoorbeeld randapparatuur zoals I²C, UART en servo's te kunnen besturen moet u aanvullende licenties kopen om toegang te krijgen tot de betreffende Flowcode-bibliotheken, of deze functies zelf in C inplementeren en vervolgens in Flowcode integreren :-(

Om deze obstakels te omzeilen is de architectuur van PobDuino gebaseerd op twee ATmega328-microcontrollers. De eerste ontvangt de code afkomstig van

De belangrijkste beschikbare commando's

DIGITALREAD,Channel(4-8) → 0/1
Lezen van een digitale ingang van de µC van het Lotus-board.

DIGITALWRITE,Channel(8-15),Value(0-100)
Schrijven naar een PWM-uitgang van het servo-board van Adafruit.

ANALOGREAD,Channel(0-3) → 0-100
Lezen van een analoge ingang van de µC van het Lotus-board. In zijn praktijklessen gebruikt Stéphane deze ingang voor het lezen van sensoren om lijnen te volgen.

DISTREAD,Channel(0-3) → D(cm)
Lezen (in cm) van een afstandssensor (Sharp GP2Y0A21), verbonden met een analoge ingang van de µC van het Lotus-board.

USREAD,Channel(4-7) → D(cm)
Lezen (in cm) van een ultrasone afstandssensor van Grove (nr. 101020010), verbonden met een digitale ingang van de µC van het Lotus-board.

SERVO,Channel(0-7),Degres(0-180),Speed(0-31/32)
Positioneringscommando voor een servomotor aangesloten op een uitgang van het Adafruit-board.

MOTOR,Side(L/R),Sens(F/R),Speed(0-100)
Commando voor een motor aangesloten op het Grove I²C motorbesturingsboard.

MOVE,Sens(F/R),Dist(mm 0-30000),Speed(0-100)
Commando om de robot in een rechte lijn te laten bewegen (afstand gemeten met de codewielen/vorklichtsluizen).

ROTATE,Sens(L/R),Angle(1-360)
Commando om de robot een bocht te laten nemen (hoek gemeten met de codewielen/vorklichtsluis).

STOP
Beëindigen van alle actieve motorcommando's.

Figuur 2. Schema van het PobDuino-board.

Flowcode. Deze maakt geen gebruik van bibliotheken voor randapparaten, alleen de basisfuncties voor UART-communicatie zijn geïmplementeerd. De tweede µC is ondergebracht op een board dat compatibel is met Arduino en gebruik kan maken van alle bibliotheken en perifere hardware uit de Arduino-omgeving. Een verbinding tussen de UART's van beide µC's (met een eenvoudig protocol voor de berichtenuitwisseling) maakt het mogelijk om vanuit Flowcode vrijwel zonder beperkingen gebruik te maken van de Arduino-hulpmiddelen. Het blokschema in **figuur 1** geeft een overzicht van de verschillende functieblokken in het PobDuino-systeem.

Hardware

Het enige board dat u zelf moet bouwen is het 'PobDuino'-board. Dit bevat:

- een ATmega328-µC;
- twee vorklichtsluizen (voor de codewielen van de motoren);
- een FTDI-interface om via USB te kunnen programmeren.

Dit board interpreteert de door Flowcode geproduceerde code, maar stuurt ook commando's naar verschillende delen van de robot.

Het schema (**figuur 2**) is direct geïnspireerd op de Arduino Uno (oude versie). Het hart van het board wordt gevormd door de bekende µC ATmega328P, samen met een 16-MHz-kristal (X1) en condensatoren voor de ontkoppeling van de voeding (C5, C6 en C7).

Via connector K4 is programmeren met ICSP mogelijk, bijvoorbeeld om de Arduino-bootloader te flashen. Ook bevat het board de gebruikelijke 'shield'-connectoren K3, K9, K10 en K11. De UART-communicatie via USB is toevertrouwd aan

een ‘klassieke’ FT232RL, het enige SMD-onderdeel dat wat lastiger is om te solderen.
De andere onderdelen zult u niet in een gewone Arduino aantreffen omdat ze specifieke PobDuino-functies vervullen: IC5 is een DC/DC-convertermodule die de batterijspanning omzet naar 5 V. IC3 en IC4 zijn vorklichtsluizen waarmee de omwentelingen van de robotwielen worden gedetecteerd. De andere boards van de robot zijn in de handel verkrijgbaar. Het Seeeduino-Lotusboard bevat de tweede ATmega328P-µC en twaalf Grove-connectoren. U kunt hier dus direct sensoren uit de Grove-reeks [2] op aansluiten. Een servoboard van Adafruit [3] bestuurt de servo’s, en een Grove I²C-motordriver [4] bestuurt de DC-motoren. Meer informatie over deze boards is online te vinden.
De montage en het samenbouwen van de verschillende boards is beschreven in de handleiding (met veel foto’s) die u in het Franstalige bestand ‘Assemblage du PobDuino’ aantreft en dat van de webpagina bij dit artikel [5] kan worden gedownload. In **figuur 3** ziet u een overzicht van de verbindingen tussen de verschillende boards.

Software

De twee ATmega328P-µC’s communiceren onderling via de seriële UART-verbinding. De µC van het PobDuino-board verzendt commando’s om de robot te activeren. Deze worden uitgevoerd door de µC van het Lotus-board (na ieder uitgevoerd commando wordt een bevestiging gestuurd).
Op het PobDuino-board moet de Arduino-bootloader worden geïnstalleerd. U kunt dit zelf doen volgens het Arduino-tutorial [6], of een voorgeprogrammeerde µC kopen. Vervolgens wordt het PobDuino-board door Flowcode voorzien van de bestanden `PobDuinoLib.c` en `PobDuinoLib.h` evenals verschillende macro’s (te downloaden van [5]).
De Arduinosketch `POBDUINO_B.ino` die gebruik maakt van diverse bibliotheken, moet in het Lotusboard worden geladen. Daarmee is het board dan klaar om commando’s uit te voeren die afkomstig zijn van de seriële monitor van de Arduino-IDE of van de µC van het PobDuino board (Flowcode).
De eenvoudigste manier om in Flowcode een nieuw programma voor de robot te creeren, is met het bestand `PobDuinoVierge.fcfx`. Dit bevat alle beschikbare macro’s

Over de auteurs

Jean-Noël Lefebvre werkt in zijn bedrijf OOTSIDEBOX aan interactieve modellen, ‘proof of concept’ experimenten, prototypes en kleine speciale effecten. Ook is hij consultant op het gebied van elektronisch ontwerpen (zowel hard- als software). Daarnaast ondersteunt hij onderwijsinstellingen op het gebied van elektronica en in het bijzonder Arduino. Zie www.ootsidebox.com

Stéphane Huet is bereikbaar via LinkedIn:
www.linkedin.com/in/stéphane-huet-ab552a61/

Figuur 3. De verbindingen tussen de verschillende boards.

Figuur 4. De spanningsregelaar in actie.

Figuur 5. Het inwendige van PobDuino.

Figuur 6. Behuizing van PobDuino, gebouwd met Meccano-onderdelen en stukken wit plastic.

(zie **kader**) terwijl de opties voor dit project al zijn geconfigureerd. U hoeft alleen maar uw eigen commando's toe te voegen. U kunt ook het bestand `HelloWorld.fcfx` openen, waarmee u het compileren en laden en de uitvoering door het PobDuino-board kunt testen. U stuurt hiermee de tekst 'Hello World' over de seriële verbinding via de USB-poort. Om te controleren of dit goed werkt hebt u een serieel terminalprogramma nodig, bijvoorbeeld TeraTerm of de seriële terminal van de Arduino IDE. De communicatiesnelheid moet worden ingesteld op 9600 baud. Na het bestand in PobDuino te hebben geladen moet de robot worden gereset door op de 'RESET'-knop van de robot te drukken of door de seriële terminal aan te sluiten (automatische reset). De (Franstalige) tekst 'Hello World' zal vervolgens op de terminal worden weergegeven. Het bestand 'Programmation de PobDuino dans Flowcode.docx' dat van [5] kan worden gedownload, beschrijft de eerste stappen met Flowcode en bevat details over de verschillende beschikbare commando's.

Als deze commando's voor u niet voldoende zijn, kunt u er nieuwe aan toevoegen door deze in de sketch `POBDUINO_B.ino` te programmeren en de overeenkomstige Flowcode-macro's te creëren. Dit opent de weg naar het gebruik van een breed scala aan sensoren en actuatoren voor Arduino zoals Grove van Seeedstudio: lezen van MP3, besturen van led-NeoPixels, lezen van RFID-chips en lichtsensors...

Conclusie

Dankzij het PobDuino-board met open-source software kunnen klanten van Pob Technology hun robots weer tot leven wekken. Als u geen Pob hebt is dat geen probleem: Stéphane stelt een 3D-printbestand ter beschikking waarmee u de achterkant van de originele Pob-robot kunt namaken. De rest van de behuizing kunt u samenstellen uit materialen die u nog hebt liggen (**figuur 6**). U kunt ook andere bestaande mechanische platforms gebruiken (bijvoorbeeld Pololu). Tot slot bevat de voor leraren bedoelde Franstalige map 'Initiation programmation' (zie [5]) zeven programmeer-lesmodules voor leerlingen van het tweede jaar ingenieurswetenschappen. Stéphane wil de modules op deze manier met zijn collega's delen.

Stéphane bedankt het Louis Aragon Lyceum voor de financiering van dit project, en het technologisch platform IDPRO (met name Marc Benbahi) voor het beschikbaar stellen van 3D-printers [7].

(170439)

Weblinks

[1] Ontdek PobDuino: hackaday.com/2017/07/27/pobduino-makes-the-most-of-grove en hackaday.io/project/26009-robot-pobduino

[2] Grove-systeem: wiki.seeedstudio.com/Grove_System/

[3] Servo-board 'Adafruit 16-Channel 12-bit PWM/Servo Driver - I2C interface - PCA9685': www.adafruit.com/product/815 & learn.adafruit.com/16-channel-pwm-servo-driver/downloads

[4] Motorbesturingsboard 'Grove - I2C Mini Motor Driver': wiki.seeedstudio.com/Grove-Mini_I2C_Motor_Driver_v1.0/

[5] Webpagina bij dit artikel: www.elektormagazine.nl/170439

[6] Installatie van de Arduino-bootloader: www.arduino.cc/en/Tutorial/ArduinoISP

[7] IDPRO: www.plateforme-idpro.fr

productronica
fast forward
powered by Elektor

the startup platform

LAUNCH YOUR STARTUP

AT PRODUCTRONICA 2019

p-ffwd 2019 – Doe mee!
12 - 15 november 2019
Messe München

Meer info:
www.elektormagazine.nl/p-ffwd

Productronica Fast Forward is brought to you by

productronica

elektor
INNOVATION > STARTUP > TRADE

Nieuw bij Android Things

C++ als booster

Tam Hanna (Slowakije)

De vorige keer dat we in Elektor schreven over Android Things [1][2], was het besturingssysteem nog volop in ontwikkeling. Google heeft het werk intussen afgesloten en biedt een API-stabiele versie aan. In dit artikel werpen we een korte blik op de belangrijkste veranderingen. Als host gebruiken we, zoals altijd, een Raspberry Pi 3 van de eerste generatie. We gaan ervan uit dat u over een Google-account beschikt.

Er is veel veranderd in Android Things 1.0: naast een nieuw permissiesysteem en veranderingen in de GPIO-API heeft Google van de gelegenheid gebruik gemaakt om een C++-programmeerinterface te introduceren en het update-systeem te verbeteren.

Voor de volgende stappen hebben we een Raspberry Pi 3 nodig met een muis, toetsenbord, beeldscherm en internettoegang. We gaan uit van de onder [3] beschikbare managementconsole voor Android Things. Daarmee kunnen we apparaten con-

figureren en images samenstellen [4]. Het is aan te bevelen om voor de volgende experimenten een nieuw project aan te maken in het Android-Things-console.
Ga naar het door Google automatisch aangemaakte model om een build-configuratie samen te stellen. Het “Modelbeheer” is bedoeld voor ontwikkelaars die meerdere hardware- of softwareversies willen beheren in één projectcategorie.
Net als in de voorgaande artikelen neemt het aanmaken van de configuratie wat tijd in beslag. Download daarna in elk geval het development-image en installeer het op de gebruikelijke manier met behulp van een SD-kaart [1].

Nieuwe GPIO-API

Open Android Studio en maak op de gebruikelijke manier een nieuw project aan. Moderne versies van de IDE (vanaf V3.2) hebben in de projectgenerator een keuzemogelijkheid om een Android Things-project te genereren. Kies als target-systeem Android Oreo en maak een Activity van het type Android Things Empty Activity aan.
Als de code hardware moet aansturen, zijn bij releaseversies van Android Things de juiste permissies nodig. Voor onze experimenten met de I/O-pennen volstaat het om in *AndroidManifest.xml* de volgende rechten in te stellen:

```
<xml version="1.0" encoding="utf-8"?>
<manifest . . .>
   <uses-permission android:name="com.google.android.
    things.permission.USE_PERIPHERAL_IO" />
```

Figuur 1: De golfvorm ziet er niet erg stabiel uit.

Android Things bewaakt de GUI-thread op dezelfde manier als normale versies van het besturingssysteem. Maak een van Thread afgeleide klasse, die in de `onCreate`-methode de *MainActivity* op gang brengt.

Net als in de eerdere artikelen willen we een blokgolf genereren om een indruk te krijgen van de uitvoeringssnelheid en de stabiliteit in de tijd. Voor het genereren van de karakteristieke golfvorm is de volgende code nodig:

```
public void run() {
    Gpio myGPIO0;
    try {
        PeripheralManager manager =
PeripheralManager.getInstance();
        myGPIO0 = manager.openGpio("BCM2");
```

Het initialiseren van de hardware-elementen gaat bij Android Things 1.0 anders in zijn werk dan bij zijn voorganger. In plaats van de PeripheralManagerService is er een volwaardige Platform-Service, die op dezelfde manier werkt als de andere system services van Android. Dat wil zeggen dat de gebruiken instantie ontstaat via `getInstance()`.
De rest van de functie ziet eruit zoals we gewend zijn:

```
        myGPIO0.setDirection(Gpio.
DIRECTION_OUT_INITIALLY_LOW);
        while(1==1){
            myGPIO0.setValue(true);
            myGPIO0.setValue(false);
        }
    }
    catch (Exception e) { }
}
```

Op een Raspberry Pi 3 die op 1,2 GHz draait, zien we bij het aansluiten van een modulatie-domein-analyser het resultaat dat is weergegeven in **figuur 1**.

Figuur 2: Deze checkbox activeert de C++-ondersteuning.

En nu met C++

Wie is opgegroeid met embedded-Java-systemen zoals MicroEJ denkt nu instinctief aan problemen met de JNI-implementatie. Die afkorting staat voor *Java Native interface*: een systeem waarmee programma's die draaien in de Java-VM native methods mogen aanroepen. Het spreekt vanzelf dat het uitwisselen van parameters tussen Java-code en native code bepaald niet snel gaat. Daarom biedt Google ook een C++-API aan, die deze problemen in theorie omzeilt. Misschien lukt het ons om het genereren van het blokgolfsignaal daarmee wat sneller te maken.

Ontwikkelen in C++ maakt geen deel uit van de standaard-distributie. Open in Android Studio de installatie-wizard onder *Tools → SDK Manager* en ga naar de rubriek *SDK Tools.* Vink daarna de entry *NDK* aan om de extra componenten te downloaden (in totaal ca. 1 GB groot).

Met Android Studio kunnen we C++-ondersteuning toevoegen aan al aanwezige projecten. Vooral bij Solutions die maar weinig code bevatten is het beter om die moeite niet te nemen en gewoon te starten met een nieuw projecttemplate. Vergeet daarbij niet om de C++-checkbox aan te vinken; alleen dan gebruikt Android Studio een projecttemplate dat geschikt is voor de C++-compiler (**figuur 2**).

Een C++-project onderscheidt zich van andere projecten doordat een extra stap is opgenomen in de assistent. In het veld C++-standaard moet u de *Toolchain Default* selecteren om de al beschikbare compiler te kiezen en extra problemen bij het inrichten te voorkomen. Na het aanklikken van Finish maakt Android Studio op de gebruikelijke manier een nieuw project aan dat, naast de al bekende Java-elementen ook een native gedeelte bevat.

Firefox nog steeds ongewenst!

Als zich bij het werken met Android Things-console "onverklaarbare" verschijnselen voordoen, dan is het zaak om de gebruikte webbrowser te controleren. Google heeft altijd weer problemen met Firefox, Edge en Safari; de ervaring leert dat hun eigen Chrome het beste werkt.

Nieuw gedownloade NDK-installaties zijn dikwijls niet compleet. Voer in de eerste stap een hersynchronisatie en/of compilatie van de applicatie uit. Als Gradle ontbrekende componenten detecteert, vraagt het in een foutmelding om die te downloaden. Meestal is het voldoende om de link in de foutmelding aan te klikken; als er een Internetverbinding is, zorgt Android Studio voor de rest.

Omdat de C-API van Android Things op het moment dat we dit schrijven nog niet definitief is, maakt die nog geen deel uit van de NDK. Open in plaats daarvan de URL [5] in een browser en download de meest recente release. Op dit moment luistert het bestand naar de naam *native-libandroidthings-1.0.zip*. In het archief vindt zich een map met de naam *native-libandroidthings-1.0*, die de gecompileerde Android Things-API bevat.

We moeten die verplaatsen naar de map die het bestand *CMakeLists.txt* bevat. Dat is een configuratiebestand dat het compileren van de native gedeelten van het project bestuurt. Op het werkstation van de auteur heet het project *ElektorCGPIO* dus het pad is *ElektorCGPIO/app*.

U moet nu Android Studio opnieuw opstarten om het project te actualiseren: de IDE is niet zo goed in het herkennen van veranderingen in het bestandssysteem.

Klik vervolgens op *External Build Files → CMakeLists.txt* om het Make-bestand voor het native gedeelte te laden. Dat is nodig, omdat C en C++-projecten vanaf een bepaalde grootte niet meer met de hand, maar met behulp van een Makefile worden gecompileerd. Dat is een soort besturingsbestand, dat het compilatietool *make* vertelt welke stappen er nodig zijn voor het bouwen van de software.

Voor het inbouwen van onze bibliotheek zijn veranderingen aan de structuur van het project nodig. Ten eerste moet u onder de regel `cmake_minimum_required(VERSION 3.4.1)` een door Google vastgelegde passage invoegen. Daarbij moet u het pad aanpassen aan de situatie op uw eigen PC:

```
set(CMAKE_MODULE_PATH $ /home/tamhan/Desktop/
stuff/2018September/ElektorATINews/ElektorCGPIO/app/
native-libandroidthings-1.0)
find_package(AndroidThings REQUIRED)
include_directories($)
```

Het doel van dit, op het eerste gezicht verwarrende, stukje code is om het module-pad van het tool uit te breiden met de map waar de eerder gedownloade en uitgepakte componenten zich bevinden. CMake zoekt tijdens het compileren van een programma in verschillende directory's naar native bibliotheken, die naar behoefte worden meegenomen.

Daar bevindt zich nog een include-bestand, dat zorgt voor het meenemen van de bibliotheek in het verdere compilatieproces. Verder naar beneden staat een structuur, die de te laden bibliotheken weergeeft in een lijst. Ook die moet worden aangepast om Android Things te kunnen gebruiken:

```
target_link_libraries(
        native-lib
        ${ANDROIDTHINGS_LIBRARIES}
        ${log-lib})
```

Het compileren van het native deel van een Android Studio-project wordt niet alleen in *CMakeLists* geconfigureerd. Ga naar

Figuur 3: Android Studio geeft de naam van de ontbrekende functie aan.

het bestand *build.gradle* van de module *App*, en voeg in het gedeelte Android een NDK-blok in:

```
android {
    compileSdkVersion 27
    defaultConfig {
        ndk {
            abiFilters 'armeabi-v7a', 'x86'
        }
        . . .
```

Android Studio en Android Things zijn, omdat ze op Java gebaseerd zijn, op applicatieniveau onafhankelijk van de CPU-architectuur. Maar onze API is een voorgecompileerde module die beschikbaar is in de vorm van assemblercode. We moeten ons daarom beperken tot de processorarchitecturen die op ARM en x86 zijn gebaseerd. Google ziet het nut niet meer van in om andere processoren zoals MIPS nog te ondersteunen
Om met native bibliotheken te kunnen werken, zijn ook veranderingen in de *MainActivity* nodig. Om te beginnen is er nu een static constructor:

```
public class MainActivity extends Activity {
    static {
        System.loadLibrary("native-lib");
    }
```

Static constructors zijn een speciale versie van de normale constructors, die worden aangeroepen bij het starten van een programma. Ze maken gebruik van de functie LoadLibrary om tijdens de uitvoering van het programma een koppeling met de in C geschreven bibliotheek tot stand te brengen.
Dat deze koppeling pas bij het uitvoeren wordt gemaakt, heeft belangrijke consequenties: de compiler voert bij het samenstellen van een gecombineerd C/Java-project geen enkele controle uit of de aan de Java-kant gedeclareerde methoden aan de C++-kant ook werkelijk zijn geïmplementeerd. Daarom is het heel belangrijk om op JNI gebaseerde code zo uitgebreid mogelijk te testen om irritante run time errors te voorkomen.
Het eigenlijke aanroepen van de native methoden gaat als volgt:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    . . .
    TextView tv = (TextView) findViewById(R.
id.sample_text);
    tv.setText(stringFromJNI());
}
public native String stringFromJNI();
```

Het is belangrijk dat native methoden in Java worden gedeclareerd met het sleutelwoord native. De regel vertelt de compiler dat de ontwikkelaar bezweert die methode te hebben gemaakt. Als dat niet het geval is, ontstaat er een fout tijdens het compileren.

Het gaat om de naam!

Java en C zijn programmeertalen die heel verschillend werken. Terwijl Java werkt met beheerd geheugen en voor strings en dergelijke is uitgerust met complexe abstracte klassen, programmeert de C-ontwikkelaar dicht bij de hardware.
Omdat we hier uit ruimtegebrek geen complete inleiding in JNI kunnen geven, verwijzen we naar de tutorial onder [6].
Maar in de praktijk is het “aanpassen” van de datatypen niet het grootste probleem van de ontwikkelaar. Wat veel lastiger is, is het vinden van de juiste syntax. Kijk bijvoorbeeld maar naar de ingewikkelde benaming voor de eerder gebruikte methode stringFromJNI in de C-code:

```
extern "C" JNIEXPORT jstring JNICALL
Java_com_tamoggemon_elektorcgpio_MainActivity_
stringFromJNI(
        JNIEnv *env,
        jobject /* this */) {
    std::string hello = "Hello from C++";
    return env->NewStringUTF(hello.c_str());
}
```

Naast de diverse vlaggen, die aangeven dat de methode moet worden geëxporteerd, vinden we hier ook de string *Java_com_tamoggemon_elektorcgpio_MainActivity_stringFromJNI*. Dat is een door de Java-compiler gegenereerde naam onder welke hij de methode verwacht te vinden.
Er is handige workaround om bij het werken met Android Studio de juiste naam van de methode te vinden. Open *MainActivity.java* en pas die aan volgens het volgende schema:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
```

Figuur 4: Ook met C++ wordt het programma niet veel sneller.

```
    setContentView(R.layout.activity_main);
    toggleWaveform();
}
public native void toggleWaveform();
```

De hier weergegeven code roept de methode *toggleWaveform* aan. Deze is nog niet gedefinieerd en daarom zullen er bij het uitvoeren van de code op de Raspberry Pi 3 fouten ontstaan. Klik nu tòch op Play om het programma via ADB te laten overzenden.
Het programma zal hooguit even opflitsen op het beeldscherm en daarna komt u weer in de Android Things-launcher. Ga dan in Android Studio naar de rubriek LogCat en zet in de combobox het filter op *Error*, zodat Android Studio alleen de foutmeldingen laat zien. Het scherm ziet er dan uit zoals in **figuur 3**.
Met de op die manier verzamelde informatie kunnen we een basisstructuur voor de methode vormen:

```
#include <jni.h>
#include <string>
#include <pio/gpio.h>
#include <pio/peripheral_manager_client.h>
extern "C" JNIEXPORT void JNICALL
Java_com_tamoggemon_elektorcgpio_MainActivity_
toggleWaveform(
        JNIEnv *env,
        jobject /* this */) {
}
```

De parameters `JNIenv` en `jobject` zijn bij JNI-methods altijd aanwezig, die kunt u negeren. Voor ingewikkelde gevallen is er een JNI-hulpobject, waarmee u gebruik kunt maken van de runtime-diensten. Zo levert JNI bijvoorbeeld objecten, waarmee interactie met Java-objecten of Java-variabelen mogelijk is.
De twee includes die met pio beginnen zorgen voor het meenemen van de Android Things-API.
Daarna kunnen we de eigenlijke uitvoermethode aanmaken. De code ziet er als volgt uit:

```
extern "C" JNIEXPORT void JNICALL
Java_com_tamoggemon_elektorcgpio_MainActivity_
toggleWaveform(
        JNIEnv *env,
        jobject /* this */) {
    APeripheralManagerClient* client =
APeripheralManagerClient_new();
    AGpio* gpio;
    APeripheralManagerClient_openGpio(client, "BCM2",
&gpio);
    AGpio_setDirection(gpio,
AGPIO_DIRECTION_OUT_INITIALLY_LOW);
    while(1==1)
    {
        AGpio_setValue(gpio, true);
        AGpio_setValue(gpio, false);
    }
}
```

In principe staat hier niets verrassends: we gebruiken de onder [7] in detail beschreven C++-API om een GPI/O-Pin-object op te bouwen en daarna een karakteristieke golfvorm te genereren. In theorie zou ons programma nu klaar moeten zijn om te gebruiken. Maar in de praktijk gaat het mis. De oorzaak is dat we al een programma op de Raspberry Pi hebben geïnstalleerd, dat poortpen BCM2 gebruikt. Daardoor is voor ons de toegang geblokkeerd.
De eenvoudigste manier om dit op te lossen is om een muis met de RPi te verbinden en in de rubriek *System* een resetcommando te geven. Een alternatief is om met ADB het programma op een wat “zachtzinniger” manier van de Raspberry Pi te verwijderen.

Daarna kunnen we ons voorbeeldprogramma opnieuw uitvoeren om het gedrag bij verschillende frequenties nader te bestuderen. Het analysediagram in het modulatiedomein is te zien in **figuur 4**. Daaruit blijkt dat de verbetering erg tegenvalt.
Wie zich verder verdiept in de Android Things-API (of een van de vele klachten van ontwikkelaars bestudeert), zal merken, dat de toegang tot de hardware ook onder C++ via het SYSFS-bestandssysteem van Linux werkt. Dat is een heel betrouwbare manier om met de hardware te werken, maar de performance laat te wensen over. API’s zoals WiringPi schrijven rechtstreeks in de registers van de processor. Maar dat is onder Android Things niet mogelijk, omdat onze programma’s niet met superuser-rechten kunnen werken.
Om de prestaties te verbeteren, zouden we rechtstreeks met het SYSFS-bestandssysteem kunnen werken. Dat zou de snelheid inderdaad wat verbeteren, maar in de praktijk is het de moeite niet waard.

Automatische updates?

Tegenwoordig kunnen we op alle mailinglists over computerveiligheid wel iets lezen over hacks van IoT-apparaten. Wat

LogCat

Een werkend Android-apparaat genereert voortdurend meldingen die door het besturingssysteem worden verzameld. In Android Studio, maar ook in verschillende andere ontwikkeltools, is er een venster “LogCat”, waarmee u de inhoud van de logfile kunt bekijken.

Figuur 5: Bijna alle partities zijn twee keer aanwezig, daardoor is updaten veiliger.

het allemaal nog erger maakt, is dat de gebruikers van die apparaten zelden of nooit hun software updaten.
Google wil ontwikkelaars helpen om die situatie te verbeteren. Daarom zorgt de Android Things-console ervoor dat het ontwikkelaars gemakkelijk wordt gemaakt om regelmatig updates aan te bieden. Omdat deze functie heel nuttig kan zijn, willen we hier van het nieuwe update-concept beschrijven.
Er wordt gewerkt volgens het basisprincipe van AB-updates: in **figuur 5** is te zien, dat op de SD-kaart bijna alle partities tweemaal aanwezig zijn. Binnenkomende updates gaan naar de ene partitie, terwijl de processor rustig doorwerkt met de software in de andere partitie. Als een update mislukt, kiest Android Things automatisch de werkende versie, zodat het systeem nooit volledig kan uitvallen.
Android Things-apparaten melden zich aan in een zogenaamd *Channel*: Google heeft vier van die channels gedeclareerd; nieuwe apparaten beginnen hun leven in het *stable-channel*. Voor een verandering van kanaal is een *Factory Reset* nodig, waarbij alle data van de gebruiker wordt gewist.
Eén van de vernieuwingen van Android Things 1.0 is een volledige API, waarmee de ontwikkelaar updates kan aangeven. Het is onder meer mogelijk om vast te leggen hoe en wanneer updates moeten worden geladen. Dat lijkt misschien bemoeizuchtig, maar het is wel belangrijk: het zou anders kunnen gebeuren dat door een update een lang reeks van verzamelde meetwaarden verloren zou gaan.
Uit ruimtegebrek kunnen we hier niet verder ingaan op de update-API; zie [8] voor een praktische implementatie

Conclusie

Het is voor ontwikkelaars vervelend dat er in Android Things 1.0 kleine veranderingen zijn aangebracht in de GPIO-API. Het zou mooi geweest zijn, als we bestaande code op de een of andere manier hadden kunnen hergebruiken.
Verder versterkt Google met deze nieuwe versie een trend die we al bij eerdere versies zagen: maximale performance is voor Google niet belangrijk, zeker niet als die, zoals bij rechtstreekse toegang tot de hardware, ten koste zou gaan van de veiligheid van het systeem.
Het besturingssysteem is een “comfortabele zandbak-achtige speelomgeving” voor iedereen die met weinig moeite gebruikersinterfaces voor dingen wil maken. Wie op zoek is naar een real time-besturingssysteem, is bij Android Things, ook in de definitieve versie, aan het verkeerde adres. ◀

180305-03

Weblinks

[1] Android op de RPi (1), Elektor maart/april 2017: www.elektormagazine.nl/160361
[2] Android op de RPi (2), Elektor mei/juni 2017: www.elektormagazine.nl/160369
[3] Android Things management-console: https://partner.android.com/things/console
[4] Documentatie Android-Things-console:: https://developer.android.com/things/console
[5] C-API: https://github.com/androidthings/native-libandroidthings/releases
[6] JNI-Tutorial: https://developer.android.com/ndk/guides/concepts
[7] Beschrijving van de C++-API: https://developer.android.com/things/sdk/pio/native
[8] Update-API: https://developer.android.com/things/sdk/apis/update

(Bijna) alles wat u altijd al wilde weten over...
behuizingen

foto: Bopla Gehäuse Systeme GmbH

Thijs Beckers

Een behuizingen voor je PCB, je kunt er maar het beste een van tevoren uitkiezen en daarna pas de print ontwerpen. Dan past het aan het eind tenminste fatsoenlijk. Hier enkele overwegingen die je kunnen helpen bij het uitkiezen van je behuizing, zodat je je schakeling een passend onderdak kunt geven.

Wanneer je met elektronica bezig bent, heb je, of dat zou je tenminste wel moeten hebben, altijd in het achterhoofd dat elektriciteit en componenten met een zeker respect behandeld dienen te worden. Componenten kunnen erg gevoelig zijn voor (statische) spanningen en stroom kan in het ergste geval dodelijk zijn. Een (te) vochtige omgeving is meestal geen goede combinatie, dat weet vrijwel iedereen. Maar te warm, zeker wanneer elektronica zich in een gesloten behuizing bevindt, kan net zo schadelijk zijn. Langzaam zien we zo al een aantal voorwaarden ontstaan waaraan een elektronicabehuizing dient te voldoen.

V *In welke omgeving komt de behuizing te staan?*

A De functie van een behuizing is in de eerste plaats het beschermen van haar inhoud tegen de factoren en elementen die op de plaats van gebruik een goede functie zouden kunnen verhinderen. Voor bescherming tegen mechanische invloeden en tegen vocht kijk je naar de IP-normering (Ingress Protection — bescherming tegen binnendringing, zie de tabellen). Overigens wordt aan de andere kant van de Atlantische oceaan hiervoor vaak de NEMA-norm (National Electrical Manufacturers Association, [1]) gehanteerd.

V *Van welk materiaal moet de behuizing zijn?*

A Grof genomen zijn er twee keuzes: metaal of plastic. Bij fijne elektronica wordt vaak gekozen voor plastic omdat dat vaak de meeste voordelen biedt, zoals elektrische isolatie en prijs. Maar in industriële toepassingen wint vaak de superieure mechanische stevigheid van metaal het. Daarbuiten kun je ook denken aan eigenschappen als warmtegeleiding, vuurvastheid, bestendigheid tegen olie, zuren en basen, roest, corrosie, krasvastheid en temperatuur waarin het materiaal stabiel blijft.
Nog een andere tak van sport zijn de biologisch afbreekbare plastics. De ASTM [2] stelt hiervoor richtlijnen op waaraan een materiaal dient te voldoen. “Biologisch afbreekbaar” kan anders nogal ruim geïnterpreteerd worden; na 1000 jaar is ‘gewone’ plastic immers ook afgebroken.

V *Hoe moet de behuizing uitzien?*

A Het oog wil ook wat en bij het uiterlijk van een behuizing spelen een aantal factoren mee. Over smaak valt natuurlijk te twisten, maar commercieel gezien is het uiterlijk in sommige gevallen misschien nog wel belangrijker dan de ergonomie. Voor je eigen schakeling is een uniek, professioneel gestileerd ontwerp misschien wat te prijzig en kies je beter bij een ‘standaard’ behuizing.
Wanneer je er iets meer van wil maken, zijn er ook tal van bedrijven te vinden waar je je eigen frontplaat naar wens kunt laten fabriceren. Kost iets meer, maar ziet er wel gelijk een stuk professioneler uit. Als je meer een doe-het-zelver bent, moet je beslist eens een kijkje nemen in een “Fablab”. Mocht je nog niet bekend zijn met dit begrip: leden krijgen toegang

tot allerlei werkplaatsen/ateliers met (professionele, industriele) apparatuur en kunnen daar naar hartenlust bouwen en creatief zijn.

Meestal specifiek voor de toepassing

Als we wat dichter bij huis kijken en onze eigen schakelingen als voorbeeld nemen, kunnen we wat specifieker zijn bij de eisen aan een behuizing. Kijken we bijvoorbeeld naar de *Verbeterde stralingsmeter* uit Elektor november 2011 [3], dan zien we dat de sensorprint een afgeschermde en lichtdichte behuizing dient te krijgen, die ook geopend moet kunnen worden voor het plaatsen van een sample. De afscherming vraagt dus om een kooi van Faraday. Een metalen behuizing ligt dan voor de hand. Metaal laat ook geen licht door, dus aan de tweede eis is ook voldaan. Verder werkt de schakeling op een lage spanning met weinig stroom, dus qua veiligheid zijn er geen bijzondere eisen. Onze keus voor het prototype: een koekblik.

Iets recenter is de *Nieuwe Nauwkeurige Nixie-klok* uit Elektor mei/juni 2016 [4]. Hierbij vonden we het belangrijk om zowel de elektronica als de Nixie-buizen in de spotlight te zetten. Om de GPS-ontvanger te laten functioneren, is een metalen behuizing uitgesloten. Aan beide eisen wordt voldaan door een doorzichtige behuizing van acryl.

Een ander voorbeeld is de Nixie-bargraph-thermometer uit Elektor jul/aug 2018 [5]. Ook hier vonden we het uiterlijk erg belangrijk. Om een stilistische temperatuurschaal te maken, hebben we ook hier voor acryl gekozen en gebruik gemaakt van de intrinsieke lichtgeleidende eigenschap van het materiaal.

Aan bovenstaande voorbeelden kunnen we zien dat de eisen aan een behuizing per project sterk kunnen variëren. Daarom geldt niet voor niets: begin met de behuizing en ontwerp de inhoud zo, dat het past! ◀

(180460)

Weblinks

[1] www.nema.org

[2] www.astm.org

[3] www.elektormagazine.nl/magazine/elektor-201111

[4] www.elektormagazine.nl/magazine/elektor-201605

[5] www.elektormagazine.nl/magazine/elektor-201807

IP-normering voor behuizingen:

Eerste cijfer

IP	Bescherming tegen:	Betekenis
0x	Geen bescherming	
1x	Grote voorwerpen (>50 mm)	Bescherming tegen toevallige, oppervlakkige aanraking met de hand. Beschermd tegen indringen van vaste voorwerpen > 50 mm
2x	Middelgrote voorwerpen (>12,5 mm)	Bescherming tegen aanraking met de vinger. Aanraakveilig enkel voor meetapparaten. Beschermd tegen indringen van vaste voorwerpen > 12,5 mm
3x	Kleine voorwerpen (>2,5 mm)	Bescherming tegen aanraking met een werktuig. Aanraakveilig enkel voor meetapparaten. Beschermd tegen indringen van vaste voorwerpen > 2,5 mm
4x	Spitse voorwerpen (>1 mm)	Bescherming tegen aanraking met een werktuig. Beschermd tegen aanraking met een draad. Beschermd tegen indringen van vaste voorwerpen > 1 mm
5x	Stofbescherming	Aanrakingsveilig doordat de behuizing geheel dicht is. Geen volledige bescherming tegen stof, maar wel voldoende om de goede werking niet te hinderen.
6x	Stofvrij	Aanrakingsveilig doordat de behuizing geheel dicht is. Volledige bescherming tegen stof.

Tweede cijfer

IP	Klasse	Beschermd tegen
x0	Geen	–
x1	Drupdicht Type I	vallende druppels
x2	Drupdicht Type II	vallende druppels op een 15° gekanteld apparaat.
x3	Spatdicht	besproeien (10 l/min) onder een hoek -60° tot 60°
x4	Plensdicht	besproeien (10 l/min) onder eender welke hoek
x5	Sproeidicht	bespuiten (12,5 l/min) onder eender welke hoek
x6	Waterbestendig	waterindringing indien bespoten (100 l/min) onder eender welke hoek
x7	Dompeldicht	waterindringing indien ondergedompeld (30 min op 1 m)
x8	Waterdicht	water onder opgegeven omstandigheden
x9	Vochtdicht	vochtigheidsgraad van meer dan 90% of besproeien onder hoge druk

Verkeerde fout in de schakeling!

hij doet het wel, maar anders dan gedacht ;-)

Dr. Thomas Scherer (Duitsland)

Een incorrecte schakeling heeft in het kader van de Homelab-rubriek een gaatje gevonden in de kwaliteitscontrole van Elektor — en is er doorheen geglipt. Dat is betreurenswaardig, maar er is meer aan de hand: de schakeling werkt weliswaar, maar compleet anders dan de auteur dacht. Een spannende en leerzame geschiedenis dus!

Waar hebben we het over: in Elektor september/oktober 2018 is een 'Sinus-FM-LF-versterker' gepubliceerd [1] met als ondertitel 'een pleidooi voor analoge technieken'. Elke elektronicus die iets heeft met audio of HF wordt door zo'n titel gegrepen. 'Ach,' denkt men dan, 'en hoe moet dat dan werken?' En voordat je er erg in hebt zit je er middenin.

Voorgeschiedenis

Er zijn weliswaar veel Elektor-lezers uit het postanaloge tijdperk voor wie alles dat niet uit bits en bytes bestaat een gesloten boek is, maar Elektor wordt ook door veel echte experts op het gebied van analoge elektronica gelezen. Al kort na de publicatie ontvingen we e-mails en opmerkingen, doorspekt met kritiek. En helaas — zo moeten we toegeven — hebben de critici gewoon gelijk.
Wat is er aan de hand: Elektor-lezer Hans-Norbert Gerbig had een door hem ontworpen schakeling ter publicatie aangeboden. Door een reeks ongelukkige omstandigheden kwam juist deze publicatie op het bureau van een Elektor-medewerker terecht nu eens niet echt sterk was in analoge elektronica en HF-techniek. Normaliter is dat geen probleem, want als er iets vreemds aan de hand lijkt, kun je het aan collega's vragen die op deze gebieden meer ervaring hebben. Dan kunnen de twijfelachtige aspecten met de auteur worden besproken en worden opgelost. Maar dat is hier niet gebeurd.
De betreffende medewerker kwamen enkele zaken weliswaar vreemd voor, maar door tijdgebrek kwamen zijn twijfels niet over de communicatiedrempel heen. Aangezien volgens de wet van Murphy alles wat mis *kan* gaan ook werkelijk mis *zal* gaan, werd de bijdrage niet door een collega tegen het licht gehouden. En dus werd de schakeling plus beschrijving compleet met fouten in Elektorlabs gepubliceerd. Dat had niet mogen gebeuren — maar toch gebeurt zoiets elke paar jaar wel een keer. Ons past slechts een welgemeende verontschuldiging: sorry!
Maar laten we van de nood een deugd maken: de overwegingen bij de schakeling, de achterliggende principes en de misvattingen zijn zonder meer leerzaam. Dat is waarom we alle relevante aspecten hieronder bespreken zodat deze geschiedenis een happy end krijgt!

Figuur 1. Het blokschema verduidelijkt het principe, dat echter niet werkt.

Uitgangspunten

De 'Sinus-FM-LF-versterker' zou met analoge techniek een tegenwicht moeten vormen voor de gebruikelijke schakelende audio-versterkers met hun digitale technologie en de typische PWM-signalen. Hij bestaat in feite uit een frequentiegemoduleerde HF-sinusgolfgenerator, een gewoon analoog versterker-IC en een laagdoorlaatfilter voor de demodulatie. In [1] was het blokschema van **figuur 1** afgedrukt. En wie werkelijk thuis is in audio en bovendien ook in analoge HF-techniek, die vermoedt nu al dat dit niet kan werken!
Waarom niet? Op deze vraag zijn meerdere antwoorden.

1. Een laagdoorlaatfilter kan geen HF met een symmetrische golfvorm demoduleren. Als de frequentie van het signaal ver boven de kantelfrequentie van het laagdoorlaatfilter ligt, worden de snelle veranderingen in het HF-signaal domweg uitgemiddeld. Het laagdoorlaatfilter doet wat het moet doen: het filtert HF uit en laat LF door. Dit geldt ongeacht de golfvorm. Het maakt niet uit of het sinus-

vormige, driehoekige, rechthoekige of andere signalen zijn. De modulatie doet (bijna) niets ter zake. Frequentie- en amplitudemodulatie hebben geen effect. Het op de HF gemoduleerde signaal is alleen achter een laagdoorlaatfilter te vinden als het op de een of andere manier de pulsbreedte van het HF-signaal verandert, met andere woorden als de integraal van de golfvorm wordt beïnvloed door de modulatie. Want dat is nou net de functie van het laagdoorlaatfilter aan de uitgang van een digitale PWM-versterker: hij integreert.

2. Een audioversterker-IC heeft een zeer beperkte bandbreedte. Voorbeeld LM386: volgens de datasheet komt het IC niet verder dan 300 kHz. Bij 1 MHz is de versterkingsfactor nog slechts 10 maal. Dus een audio-IC is van zichzelf dus eigenlijk al een laagdoorlaatfilter. Boven 1 MHz neemt de versterking met ongeveer 30 dB per decade af. Bij 10 MHz bedraagt die dus ongeveer 0,3 en hebben we geen HF-versterker meer maar een HF-verzwakker. *Quod erat demonstrandum*: een audio-IC kan geen HF versterken. Het bovenstaande geldt ook voor een TBA820 en andere audio-IC's. En daarom kan de schakeling van figuur 1 niet werken — zelfs als het mogelijk zou zijn een frequentiegemoduleerd HF-signaal met een laagdoorlaatfilter te demoduleren. En dat kan niet, zoals bij punt 1 aangetoond.

Ergo klopt de basisaanname van de schakeling van twee kanten niet. Een sinus-FM-LF-versterker kan niet werken, ongeacht hoe u het wendt of keert.

Figuur 2. De eigenlijke schakeling van de sinus-FM-LF-versterker.

En toch versterkt de schakeling!

Werp nu eens een blik op de schakeling van **figuur 2** en bouw die op, sluit een audiosignaalbron op de ingang aan en een luidspreker op de uitgang en zet er pakweg 5 V op, dan hoort u tot uw verbazing geluid. Hè!?

De versterker versterkt echt. Een goede elektronicus zal dat vrij snel zien. Maar omdat theoretische overwegingen pas steekhoudend zijn wanneer ze door metingen worden gestaafd, heb ik de schakeling opgebouwd en het onderstaande door metingen 'bewezen'. Daarover later meer.

Aan de linkerkant is rond de transistor een kristaloscillator opgebouwd. Kennelijk moet een laagfrequent signaal op de collectorweerstand van de transistor de gegenereerde frequentie moduleren. De gemoduleerde HF moet dan van de emitterweerstand worden afgenomen en via een potentiometer van 10 kΩ naar de ingang van het audioversterker-IC worden doorgesluisd. De basisversterking kan nogmaals worden gewijzigd met een andere potentiometer van 10 kΩ (aan pin 8) en wel tussen 20 en 200 maal. Vervolgens komt er aan de rechterkant een laagdoorlaatfilter met een 47-µH-spoel en een 1-µF-elco. De elco van 470 µF is de normale uitgangscondensator en dient om de gelijkspanningscomponent van het uitgangssignaal te scheiden. Met de 25-kΩ-potentiometer bovenin het schema wordt zoeits als het werkpunt van de FM-sinusgenerator ingesteld. Tot zover de beschrijving van de schakeling zoals die bedoeld was.

Maar waarom versterkt de schakeling dan? Zelfs als de oscillator oscilleert en inderdaad in frequentie gemoduleerd kan worden, komt er toch een aanzienlijke hoeveelheid LF op de ingang van het versterker-IC terecht. Voor LF is de effectieve weerstand van de combinatie van transistor en basisweerstand gelijk aan het quotiënt van die weerstand (10 kΩ) en de stroomversterking (typisch 100 voor een BF494), dus ongeveer 100 Ω. Het signaal dat op de collectorweerstand wordt gezet, verschijnt dan (met de genoemde waarden) met een factor 0,8 'versterkt' bij de 10-kΩ-potentiometer. Aan het hete einde staat daarom op zijn best een gesommeerde mix van HF en LF. En deze mix komt onverkort op de ingang van het audioversterker-IC terecht. De LF wordt op geen enkele manier door zoiets als een hoogdoorlaatfilter weggefilterd. En het IC is, zoals we al opmerkten, zelf al een laagdoorlaatfilter, dus in principe wordt alleen het LF-signaal versterkt. Het laagdoorlaatfilter aan de uitgang heeft nagenoeg niets meer te doen. U zou het dan ook gewoon weg kunnen laten.

Om een lang verhaal kort te maken: de versterker versterkt in deze schakeling omdat het een bijna normale audioversterker is, opgeleukt met een een handjevol nutteloze componenten. Je zou alles links van de 10-kΩ-pot (dat is die aan pin 3), evenals de potmeter van 25 kΩ, de weerstand van 68 Ω en het laagdoorlaatfilter, weg kunnen laten en een versterker overhouden die het zelfs nog beter zou

Figuur 3. Mijn ‘quick&dirty’ prototype van de schakeling van figuur 2.

doen. De sinusgenerator heeft domweg geen enkele functie.
Maar omdat de versterker ondanks de HF-componenten versterkt, concludeerde de auteur dat zijn idee correct was.

En ‘in het echt’?

Om te voorkomen dat iemand zegt dat het bovenstaande alleen theorie is, heb ik het zekere voor het onzekere genomen en de schakeling volgens het schema van **figuur 2** op gaatjesprint opgebouwd. Ik krijg daar zeker geen schoonheidsprijs voor, zoals u kunt zien in **figuur 3**, maar het werkt wel. In plaats van een kristal van 18 MHz heb ik er een gebruikt van 10,24 MHz uit mijn eigen junkbox. Ik had ook nog een BF494 liggen, alleen de LM386 moest ik bestellen. Een eerste meting aan de collector van de transistor liet de veronderstelde HF met een niveau van ongeveer 55 mV_{eff} en de verwachte frequentie zien — dus de generator oscilleert werkelijk. De HF-amplitude kan in zekere mate worden ingesteld met de potentiometer van 25 kΩ.

Figuur 4. Het 10,24-MHz-signaal. Er is geen enkele modulatie te bespeuren.

Figuur 4 toont de HF na het aansluiten van een audio-signaalbron met 600 Hz en een behoorlijke amplitude. Zoals de auteur al stelt zien we geen amplitudemodulatie, maar helaas (tenminste met het blote oog) ook geen frequentiemodulatie. Ook niet met een andere tijdbasis. Als we met de tijdbasis in de richting van LF gaan, zie we (**figuur 5**) een signaalmix van 10,24 MHz, ‘gepiggybackt’ op het aangelegde ingangssignaal van 600 Hz. Iets anders kan het eigenlijk niet zijn.
Aan de uitgang verschijnt zoals verwacht het ‘versterkte’ 600-Hz-signaal bij de luidspreker. De storingen in **figuur 6** zijn te wijten aan mijn provisorische opbouw van de schakeling en heeft niets te maken met het principe daarvan.
Zoals u ziet, stemmen theorie en praktijk hier perfect overeen.

Bel voor de tweede ronde

Dat is echter nog niet alles. Er zijn bij de schakeling van figuur 2 nog enkele details waaraan we kunnen zien hoe het eigenlijk niet moet, want het is zeker geen schoolvoorbeeld van ‘best practice’. Het begint met een kleinigheid die echter grote gevolgen heeft: het laagdoorlaatfilter met zijn 68-Ω-weerstand en 100-µF-elco rechtsboven is dik in orde, maar dat de voedingsstroom van de LM386 hier ook doorheen loopt, is onzin. Integendeel, de versterker clipt daardoor al bij een relatief gering signaalniveau. Wanneer u er meer dan enkele tientallen mW uit wilt halen, dan is het beter om pin 6 van het IC rechtstreeks aan de voeding te leggen.

Het feit dat de massa van de signaalbron niet aan de massa van de versterker ligt, maakt de schakeling bijna onbruikbaar. Mijn exemplaar van de schakeling werkte om die reden aanvankelijk niet, omdat de massa van mijn functiegenerator niet potentiaalvrij was, net zomin als de uitgang van mijn labvoeding waarop ik de schakeling had aangesloten. Zodoende sloot mijn functiegenerator de bovenkant van de collectorweerstand naar massa kort. En daarom was er ook geen LF of meetbare HF bij de emitterweerstand te vinden. Twee condensatoren van 1 µF in serie met de klemmen van de functiegenerator losten dat probleem op, maar daarmee kreeg ik wel zichtbare storingen die via de zo ontstane aardlus geïnjecteerd werden, zoals in figuur 6 te zien is. Laat dat een les zijn: verander nooit

de referentiemassa van signalen tenzij het echt niet anders kan.
Driemaal is scheepsrecht: zelfs met de twee condensatoren was er nog geen geluid te horen. Draaien aan de 25-kΩ-potentiometer bovenin figuur 2 leverde soms (bij grote weerstandswaarde) iets dat op een zacht toontje leek. Wat is hier aan de hand? De transistor heeft om te oscilleren en het doorgeven van het LF-signaal een minimale stroom nodig. Die veroorzaakt echter een spanningsval over de emitterweerstand. Afhankelijk van de stand van de erop volgende 10-kΩ-potentiometer zal een paar honderd mV echter al tot gevolg hebben dat de uitgang van het versterker-IC tegen de positieve voedingsspanning vastloopt. Dan versterkt hij niet meer. Dat komt omdat deze gelijkspanning niet van pen 2 gescheiden is. Er zou een condensator tussen de collector van de transistor en het warme einde van de 10-kΩ-potentiometer moeten zitten. Nadat ik die condensator had gemonteerd, kon ik de optimale stroom voor de maximale HF-amplitude met de 25-kΩ-potentiometer instellen zonder het IC door een offsetspanning te laten vastlopen.
Tot slot kon ik in een prettig telefoongesprek met de heer Gerbig duidelijk maken dat het hier niet om een zogenaamde 'meegetrokken oscillator' gaat. Zelfs als het er een zou zijn, dan zou het niet mogelijk zijn om die op deze manier met LF — een spectrale frequentie-mix — aan te sturen om de oscillatorfrequentie te moduleren. Men kan het signaal van de oscillator op FM en kortegolf ontvangen, dus er is wel een beetje FM- en ook AM-modulatie, maar de modulatiediepte is erg gering en is veeleer een artefact ten gevolge van de niet-lineariteit van T1.

Conclusie

Je kunt bij analoge elektronica veel verkeerd doen. Ervaring en door collega's bevestigde vuistregels zijn erg handig als u niet het risico wilt lopen dat iets het niet doet. En dat gebeurt maar al te gemakkelijk. De sinus FM-LF-versterker is in dit opzicht geen slecht voorbeeld. En wees eerlijk: wie heeft er nog nooit geblunderd bij de ontwikkeling van een schakeling? Ik kan daar zelf over meepraten, wat zeg ik: een artikel over schrijven. Maar je leert van je fouten. De hier beschreven fouten zullen in elk geval nooit meer gemaakt worden! ◀

180589-03

Figuur 5. Het audioversterker-IC krijgt een mix van een signaal van 600 Hz met daarop gesuperponeerd een signaal van 10,24 MHz (en laat de HF-component verdwijnen).

Figuur 6. Het iets versterkte 600-Hz-signaal aan de uitgang van de schakeling. Het HF-signaal is al voor het laagdoorlaatfilter verdwenen.

Links and literature

[1] Artikel 'Sinus-FM-LF-versterker':www.elektormagazine.nl/180275-02

[2] Literatuur: Otto Liman, 'Funktechnik ohne Ballast', Franzis-Verlag, 12e druk, p. 148/149.:

FM-radio met RDS

een hoed voor de Raspberry Pi

Fabian Bugelmüller, Christoph Fornezzi, Franz Parzer (HTL Steyr, Oostenrijk) en
Ton Giesbers, Mathias Claußen (Elektor-lab)

Een Raspberry Pi, een zelf ontwikkelde HAT met stuurprogramma's en een grafische gebruikersinterface die speciaal voor deze toepassing zijn geschreven, zijn de ingrediënten van een VHF-radio die er niet alleen interessant uitziet, maar ook uiterst instructief is dankzij de uitgebreide documentatie. Omdat je nooit uitgeleerd raakt!

PROJECT-INFO

Raspberry Pi 2 of 3 (B+)
HAT FM-radio
Touchscreen

beginners
➡ gevorderden
experts

4 uur

Gereedschap voor SMD-solderen, reflow-oven aanbevolen

€ 65

Als je op het internet naar een HAT zoekt (*Hardware Attached on Top*) die de Raspberry Pi in een FM-tuner verandert, kom je onvermijdelijk goedkope breakout-boards tegen met de Si4703-chip van Silicon Laboratories. Deze chip bevat een complete FM-radio met RDS (van antenne-ingang naar analoge stereo-uitgang) en is ideaal voor gebruik in mobiele apparaten.

Van dezelfde fabrikant is echter een soortgelijke chip met de naam Si4731 verkrijgbaar, die niet alleen betere ontvangsteigenschappen heeft, maar ook een digitale I²S-audio-uitgang heeft. Hierop kan een audioversterker met digitale I²S-ingang worden aangesloten die doorgaans veel betere A/D-converters heeft dan de Si4703.

Als *goodie* krijgt onze HAT een infraroodontvanger, zodat u de radio kunt bedienen met een standaard afstandsbediening. De GUI is een 3,5" LC-aanraakscherm.

Deze hardware zal worden aangestuurd met Linux-software, die naast de stuurprogramma's voor de gebruikte IC's ook een speciale radio-app bevat voor een luxueus-comfortabele bediening van de radio.

Onze FM-radio-HAT werkt met alle 'normale' RPi-versies vanaf model 2; het gebruik van Raspberry Pi 1 wordt niet ondersteund noch aanbevolen. De HAT is ook getest met de Zero en draait daarop — maar op eigen risico …

Wide Receiver

Het schema van **figuur 1** toont de paar componenten waaruit de FM-radio bestaat.

Si4731

De gebruikte radio-ontvangstchip is de 'Broadcast AM/FM Radio Receiver' Si4731-D60 [1] in een kleine 20-pins QFN-behuizing van Silicon Laboratories, met firmware-versie D60. De ontvanger kan niet alleen FM-radio ontvangen in de normale omroepband (64...108 MHz), maar ook AM-radio in de middengolf (520...1710 kHz) (**figuur 2**). De digitale audio-interface van de Si4731 werkt in de slavemodus en ondersteunt verschillende audio-gegevensformaten (hoogste bit eerst), waaronder I²S en links-uitgelijnde modi. De chip kan RDS decoderen en, net als de audiodata in I²S-formaat, uitvoeren met de gebruikelijke samplefrequenties van 32 kHz, 44,1 kHz en 48 kHz. De Si4731 wordt via de I²C-interface rechtstreeks vanaf de RPi geconfigureerd.

Figuur 1. Schema van de FM-radio.

Het IC heeft twee antenne-ingangen, een voor FM en een voor AM. In het geval van FM zorgt L2 voor resonantie. De suppressordiode D1 zorgt voor ESD-bescherming, C10 filtert de DC-component uit het signaal. Overigens levert een eenvoudige (dunne) draadantenne van 75 cm betere ontvangstresultaten dan een telescoopantenne. We hebben enige ervaring opgedaan met FM-ontvangst, maar hebben tot nu toe nog niet met AM geëxperimenteerd. Voor AM-ontvangst moet een ferrietstaaf- of raamantenne van 180...450 µH worden gebruikt, aangesloten op de AM-ingang van de Si4731 via een condensator van 470 nF (C11), zoals beschreven in de datasheet. Natuurlijk hangt het aantal windingen af van de eigenschappen van de ferrietstaaf. Zoek gewoon op internet naar mogelijke oplossingen voor een geschikte antenne.

C5 en C6 zijn ontkoppelcondensatoren, L1 zorgt voor een stabiele voedingsspanning. R8...R10 zijn beschermingsweerstanden zoals voorgeschreven in de datasheet.

SSM2518

Voor de audioweergave hebben we gekozen voor de SSM2518 van Analog Devices [2]. Deze klasse-D-versterker heeft een digitale seriële audio-interface die gegevens in I²S-formaat accepteert en die ook via de I²C-bus wordt bestuurd. De twee stereo-uitgangskanalen leveren 2 W in 4 Ω. De schakeling rond het IC volgt de standaardapplicatie uit de datasheet. C7 ontkoppelt de 3,3-V-voeding (*Digital Supply Voltage*, DVDD) en C1 en C8 de 5-V-voeding (*Speaker Supply Voltage*, PVDD).

Een polymeercondensator met een zeer lage ESR van slechts 30 mΩ wordt gebruikt om de 5-V-voeding voor de vermogenssectie van het IC te ontkoppelen. Volgens het gegevensblad hoeven de luidsprekeruitgangen slechts dan van filters te worden voorzien als de verbindingen naar de luidsprekers langer zijn dan 20 cm. In de praktijk is echter gebleken dat zelfs als aan deze voorwaarde is voldaan, de HF-componenten nog steeds zo sterk worden afgestraald dat er geen RDS-ontvangst meer mogelijk is. De filters (ferrietkralen L4...L7 en condensatoren C12...C15) aan de uitgangen elimineren het probleem. Overigens zijn R5 en R6 de pullup-weerstanden die nodig zijn voor de I²C-bus.

IR-ontvanger
Een 38-kHz IR-ontvanger (IC1, een TSOP4138) ontvangt, versterkt, filtert en demoduleert het signaal van een IR5-zender (ook 36 kHz, van televisies of muzieksystemen) en stuurt de informatie naar de GPIO26-lijn van de Raspberry Pi. De RPi interpreteert het infraroodsignaal en voert de bijbehorende opdrachten uit om de radio te bedienen.

EEPROM
Opdat de hardware voldoet aan de HAT-specificaties (zie verderop), moet een EEPROM met specifieke informatie voorhanden zijn. IC3 is een standaard 24C64 EEPROM die ook communiceert via de I²C-bus.

Figuur 2. Zo ziet de Si4731 er van binnen uit (bron: Silicon Laboratories).

CS2300
Bij de eerste prototypen werd een oscillator als klokgenerator gebruikt. Er ontstond echter een probleem omdat het door de oscillator gegenereerde hoofd-kloksignaal niet exact overeenkwam met die van de I²S-klok, waardoor herhaaldelijk bits verloren gingen. Aangezien de I²S-klok exact 1/256-van de systeemklok moet zijn, wordt in plaats van de oscillator een CS2300-frequentievermenigvuldiger gebruikt, die de I²S-klok met 256 vermenigvuldigt. De klokken van het systeem en I²S lopen nu jittervrij synchroon.
De CS2300-CP-CZZZ van Cirrus Logic wordt geconfigureerd en bestuurd via I²C. De schakeling is conform de datasheet. De 10-pF-condensator C16 op CLK_Out voor het dempen van de harmonischen is niet vereist en is daarom weggelaten.

Voeding
Misschien onnodig te vermelden dat, om de radio van stroom te voorzien, een voeding van goede kwaliteit moet worden gebruikt die daadwerkelijk (en niet alleen op papier) 3 A kan leveren. De voeding moet niet alleen de RPi, het display en de ontvanger voeden, maar ook de audio-eindversterker met 2 x 2 W. En hoeveel kwaliteit kunt u verwachten van een netstekervoeding die slechts drie euro kost?

Software voor de hardware

Om de hardware onder Linux te kunnen gebruiken, moeten speciale stuurprogramma's voor de afzonderlijke hardware-componenten aanwezig zijn (en geïntegreerd) in de Linux-kernel. Er zijn al stuurprogramma's voor de klasse D-versterker in de kernel, voor de radiotuner is de bestaande driver voor de Si476x-familie aangepast. Hier moesten de meeste veranderingen worden doorgevoerd, omdat het twee heel verschillende componentenreeksen zijn. Voor de frequentievermenigvuldiger CS2300 moesten ook stuurprogramma's van een andere component worden genomen en gewijzigd.
De ontwikkeling van stuurprogramma's is ingewikkeld en vereist programmeerervaring, en het is lastig om die stuurprogramma's in de Linux-kernel te integreren en zonder fouten te compileren. Maar geen paniek! We hebben een kant en klare image voor de hardware van de FM-radio samengesteld; u kunt die gewoon naar de SD-kaart van de Raspberry Pi kopiëren. Het werkt net als elke andere distributie zoals Raspbian of Kodi. De speciale toepassingssoftware piRadio is ook al in de image aanwezig.
Als u zich echter verder in de software wilt verdiepen en/of uw eigen distributie wilt samenstellen (eventueel met een andere gebruikersinterface), vindt u verwijzingen, voorbeelden, stuurprogramma's, instructies en nog veel meer in de talrijke documenten op de Elektor-projectpagina [4].

Mijn HAT is een driekanten steek...

Een HAT is een uitbreidingskaart voor de Raspberry Pi die voldoet aan de HAT-specificaties [1]. Een HAT wordt aangesloten op de 40-pins GPIO-pinheader van

Weblinks

[1] HAT-specificaties: https://github.com/raspberrypi/hats
[2] Datasheet Si4731: www.silabs.com/documents/public/data-sheets/Si4730-31-34-35-D60.pdf
[3] Datasheet SSM2518: www.analog.com/media/en/technical-documentation/data-sheets/ssm2518.pdf
[4] Projectpagina: www.elektormagazine.nl/180119-03
[5] FM-radio op Elektor Labs: www.elektormagazine.nl/labs/fm-radio-receiver-with-rds-for-raspberry-pi
[6] PiRadio op Elektor-Labs: www.elektormagazine.de/labs/piradio-for-fm-radio-receiver-with-rds-for-raspberry-pi-160520-1
[7] FM-radio onder Volumio: https://github.com/rpi-Receiver/
[8] FM-radio op Github: https://github.com/ElektorLabs/160520-FM-Radio-Receiver-with-RDS-for-Raspberry-Pi

Figuur 3. Layout van de FM-radio-HAT.

de RPI. Twee van deze 40 pennen zijn exclusief gereserveerd voor het aansluiten van een 'ID EEPROM' die informatie opslaat over het board, welke functies het board heeft, welke GPIO's moeten worden geconfigureerd en of de RPi en de HAT via de micro-USB-aansluiting van de RPi worden gevoed of via de HAT.

In principe moet een HAT aan de volgende voorwaarden voldoen:

- pricipiële add-on voorwaarden;
- geldige gegevens in het EEPROM (leveranciersgegevens, GPIO-map en device-tree);
- 40-polige busstrip;
- bij voeding via de HAT (*back-powering*) moet de RPi ten minste 1,3 A kunnen krijgen.

Niemand verplicht u echter om de print van een HAT rechthoekig te maken (**figuur 3**)! De *Hardware Attached on Top* bevindt zich op een driehoekige print (met uitsparing voor de aansluiting van het aanraakscherm). Dit ziet er niet alleen bijzonder uit, maar het spaart ook productiekosten. Het belangrijkste is dat de componentgroepen zo ver mogelijk van elkaar verwijderd zijn en logisch gescheiden zijn. Omdat de infraroodontvanger en de EEPROM niets te maken hebben met de eigenlijke functie, zijn ze duidelijk gescheiden van de andere componentgroepen. De radio-ontvanger bevindt zich aan de bovenkant van de print, de frequentievermenigvuldiger bevindt zich in het midden en de klasse D-versterker zit rechtsonder. Met de radio-ontvanger zijn de voeding, FM, AM en de periferie van de digitale uitgangen zo goed mogelijk van elkaar gescheiden als mogelijk is op zo'n kleine print. Bij de versterker zijn de filters zodanig geplaatst dat de ferrietkralen zich zo dicht mogelijk bij het IC bevinden en de condensatoren zo dicht mogelijk bij de aansluitingen voor de luidsprekers zitten. Alle ontkoppelcondensatoren zijn direct bij de betreffende IC's geplaatst. De EEPROM heeft een jumper voor schrijfbeveiliging. Pas wanneer u de jumper plaatst, kan de EEPROM worden beschreven.

De print van de FM-radio wordt op de RPi bevestigd met 17 mm lange M2,5-afstandsbussen (male/female), zie **figuur 4**. Voor de montage van het display is naast de Ethernet-aansluiting een vierde 17 mm lange afstandbus (eveneens M2,5 male/female) vereist, plus een M2,5-moer die compenseert voor de printdikte. Bevestig de radioprint op de RPi en schroef vier M2,5-afstandsbussen van 14 mm (ook male/female) in de onderste afstandsbussen. De vier afstandbussen voor het LC-aanraakscherm bevinden zich dus op dezelfde hoogte; de inwendige schroefdraad wijst naar boven. Dit reduceert de belasting van de 26-pins connector van het beeldscherm.

Als het LC-display niet is gemonteerd maar in plaats daarvan een extern aanraakscherm wordt gebruikt, worden de afstandsbussen en de moer van 14 mm verwijderd en moet de print worden vastgezet met M2,5 x 6-schroeven.

Figuur 4. Montage van de HAT op de Raspberry Pi.

Software voor de radio

Als u naar naar radio-apps zoekt, komt u al snel KRadio, Gnomeradio of RDS Surveyor tegen. Er zijn ook enkele tv-apps

Figuur 5. Het hoofdscherm van piRadio.

Figuur 6. De zender is in het geheugen opgeslagen.

Figuur 7. Rechtstreeks de frequentie intoetsen.

Figuur 8. Wilt u echt terug naar Raspbian?

die kunnen worden gebruikt om naar radio te luisteren. Elke app heeft zijn sterke en zwakke punten, maar voor gebruik met een IR-afstandsbediening en een aanraakscherm zijn ze geen van alle optimaal, omdat ze zijn ontwikkeld voor een desktopsysteem met muis en toetsenbord. Ze zijn moeilijk te gebruiken op een klein 3,5-inch aanraakscherm met geringe resolutie.

Daarom heeft Mathias Claußen van het Elektor-lab een programma ontwikkeld met de naam piRadio om de FM-ontvanger te bedienen. De software is specifiek ontwikkeld voor deze hardware en voor bediening met een 3,5”-aanraakscherm met 480 x 320 pixels (zoals het populaire model Waveshare35a). De bediening is volledig intuïtief.

Figuur 5 toont het display van piRadio. In het bovenste gedeelte (als de afkorting ‘RDS’ rechts bovenaan verschijnt), wordt de *Program Service Name* (PS, de zendernaam) links van de momentele afstemfrequentie weergegeven. De tweede regel toont de momentele RDS-radiotekst (RT). Als alleen de stationsnaam kon worden gedecodeerd, toont het display een regel met een lichter lettertype voor een betere leesbaarheid. Als er geen RDS-informatie wordt ontvangen, wordt alleen de momentele frequentie weergegeven. De ontvangstveldsterkte wordt weergegeven in een kleine balkdisplay helemaal rechts onder het antennesymbool. Hoe hoger de waarde, hoe beter de ontvangst.

In het onderste gedeelte bevinden zich acht knoppen die in eerste instantie met ‘not used’ zijn gemarkeerd. Dit is het zendergeheugen met acht knoppen. Er zijn vier van dergelijke pagina’s die u kunt omschakelen met de knop M[x], zodat u in totaal 32 zenders kunt opslaan. Om een frequentie op te slaan, gaat u eerst naar de pagina waar u het station wilt opslaan, dan gebruikt u de pijltoetsen om de frequentie te selecteren en drukt u gedurende 5 seconden op de gewenste knop op het aanraakscherm. Het display toont een ‘bladwijzer’-sterretje rechtsboven op het display, onder de RDS-afkorting (**figuur 6**) wanneer het station is opgeslagen.

Omdat het meestal erg onhandig is om de frequentie met de pijltjestoetsen te veranderen, hebben we een interface voor directe invoer geïmplementeerd (**figuur 7**). Wanneer u op het display drukt, gaat u naar het invoerscherm voor de frequentie (gebruik daarvoor op de afstandsbediening de toets ‘Menu’, op het toetsenbord de [E] toets). De momentele frequentie wordt nu weergegeven, en u kunt gewoon beginnen met het invoeren van een nieuwe frequentie, omdat bij het eerste cijfer de oude frequentie wordt gewist. Voor 98,55 MHz voert u dus [9][8][.][5][5] in. Als u een fout maakt, drukt u gewoon op de backspace-toets. Als de frequentie die u hebt ingevoerd zich binnen een geldig bereik bevindt, verschijnt de knop ‘Okay’. Als u niet tevreden bent met uw keuze, kunt u op ‘Abort’ drukken en teruggaan

naar het hoofdmenu. De instelling via het toetsenbord gaat op vergelijkbare wijze, op de afstandsbediening komt de Enter-toets met Backspace overeen, de A/B-toets fungeert als decimale punt en de OK-toets is natuurlijk 'Okay'.
Na het opslaan kunt u met een afstandsbediening of een toetsenbord toegang krijgen tot de geheugenplaatsen door op [xy] op de afstandsbediening te drukken, waarbij x de stationspagina is en y het stationsnummer. Om bijvoorbeeld het tweede station van de eerste pagina te krijgen, drukt u op 12, voor het vierde station op de tweede pagina gebruikt u 24. Voor pagina 1 hoeft u niet eerst op pagina 1 te drukken, maar kunt u de stations rechtstreeks kiezen (en een seconde wachten). Om een ander station op dezelfde pagina te kiezen, kunt u ook [0y] invoeren. Als u een stationgeheugen selecteert dat niet bestaat of waar geen station heeft opgeslagen, zal de radio de frequentie niet veranderen.
Tussen het tekstvenster en het zendergeheugen bevinden zich de al genoemde afstemknoppen (de pijltjestoetsen) aan de linkerkant, die drie seconden moeten worden ingedrukt voordat een zoekopdracht begint. Op een afstandsbediening kunt u het zoeken starten met de knoppen voor snel vooruit of terug, op het toetsenbord met de pijltoetsen links en rechts. Druk op de pijltoetsen omhoog of omlaag om de frequentie te verhogen of te verlagen.
Als u het volume wilt wijzigen, gebruikt u de bijbehorende knoppen of zet u het geluid uit/aan met de mute-knop. De knoppen op de afstandsbediening zijn hetzelfde, gebruik op een toetsenbord de plus-, min- en M-knop.

Tenslotte de aan/uit-knop: als u kort op deze knop drukt, wordt de radio gewoon uitgeschakeld; houdt u de knop daarentegen langer dan 10 seconden ingedrukt, dan verschijnt er een klein desktop-pictogram in de rechterbovenhoek van het scherm. Wanneer u de knop vervolgens loslaat, verschijnt een berichtvenster waarin u wordt gevraagd of u echt terug wilt naar het bureaublad. Als u op 'yes' klikt, wordt de toepassing gesloten en hebt u toegang tot het bureaublad (**figuur 8**). Anders wordt het bericht gesloten en blijft de toepassing actief.
Een overzicht van alle opdrachten en invoer voor het aanraakscherm en de overeenkomst van het toetsenbord en de afstandsbediening is te zien in **tabel 1**.

Ahtergrondinformatie

Zoals eerder vermeld, hebben we een op Raspbian gebaseerd image voor u voorbereid dat alle stuurprogramma's voor de radio-hardware en het LCD-scherm bevat, alsmede LIRC voor de RC5-afstandsbediening. Het Qt-framework als basis van de GUI is geïnstalleerd en het opstartbestand is aangepast. Het image is klaar en hoeft alleen nog maar naar de SD-kaart van de Raspberry Pi te worden gekopieerd om de FM-radio met alle functies te kunnen gebruiken.

Tijdens de ontwikkeling van het project zijn echter talrijke spin-offs, tekstdocumenten en programmacodes tot stand gekomen die we op de projectpagina [4] en op de twee relevante Elektor Labs-pagina's over Elektor-Labs [5] [6] voor u hebben klaargezet.

Tabel 1. Bediening van de PiRadio-software met aanraakscherm, knoppen en afstandsbediening.

Functie	Knoppen	Afstandsbediening
Volume verhogen	+	Vol+
Volume verlagen	-	Vol-
Geluid aan/uit	M	MUTE
Uitschakelen	Q	POWER
Frequentie verhogen	Pijltje naar boven	CH+
Frequentie verlagen	Pijltje naar beneden	CH-
Station zoeken (hogere frequentie)	Pijltje naar rechts	Videotext rood
Station zoeken (lagere frequentie)	Pijltje naar links	Videotext blauw
Frequentie-instelling tonen/verbergen	E	Menu
Frequentie wissen	wissen (BACKSPACE)	ENTER
Decimale punt	Punt of komma	A-B
Bevestigen van de ingestelde frequentie	ENTER	OK
Cancel en stop	ESCAPE of E	Menu

ONDERDELENLIJST

Weerstanden:
R1...R6 = 3k9, 0,125 W, 1 %, SMD 0805
R7...R10 = 100 Ω, 0,125 W, 1 %, SMD 0805

Condensatoren:
C1 = 10 µ/16 V, 20 %, 5 mm radiaal SMD, 0,03 Ω /2,2 A
C2...C8,C17 = 100 n, 50 V, 10 %, X7R, SMD 0805
C10 = 1 n, 50 V, 10 %, X7R, SMD 0603
C11 = 470 n, 16 V, 10 %, X7R, SMD 0603
C12...C15 = 470 p, 50 V, 5 %, C0G/NP0, SMD 0805
C16 = 10 p, 50 V, ±0.5 pF, C0G/NP0, SMD 0805, niet gemonteerd

Spoelen:
L1 = SLF7032T102MR13-2PF, 1mH@100kHz, 130 mA, 4,78 Ω (TDK)
L2 = LQW18ANR27G00D, 270nH@100MHz, 110 mA, 3,4 Ω (Murata)
L4...L7 = BLM18PG471SN1D, 470Ω@100MHz, 1 A, 0,2 Ω (Murata)

Halfgeleiders:
D1 = CM1213A-01SO, SMD SOT-23 (ON Semiconductor)
IC1 = TSOP4138, 3-polig bedraad (Vishay)
IC2 = SI4731-D60-GM, SMD 20-polig, QFN (Silicon Labs)
IC3 = CAT24C64WI-GT3, SOIC-8 (ON Semiconductor)
IC4 = SSM2518CPZ, LFCSP 20-polig -WQ (Analog Devices)
IC5 = CS2300-CP-CZZ, MSOP-10 (Cirrus Logic)

Diversen:
K2,K3 = 2-polige printkroonsteen, steek 3,5 mm
J8 = 2x20-polige GPIO-header voor RPi (extra lang)
4 x afstandsbus 17 mm M2,5 (male/female), bijv. Ettinger 05.12.173
4 x afstandsbus 14 mm M2,5 (male/female), bijv. Ettinger 05.12.143
5 x M2,5-moer
1x5-polige header, verticaal, raster 2,54 mm, gedeeld voor AM/FM/PW-pads
J8 = 3,5”-touchscreen-LCD, bijv. Waveshare 3.5inch RPi LCD (A)
Raspberry Pi vanaf model 2

Print 160520-1 v2.0

- De basis van de FM-radio is de afstudeerscriptie (RpiReceiver.pdf) van de auteurs Fabian Bugelmüller en Christoph Fornezzi aan de HTBLA Steyr in Oostenrijk. Deze beschrijft in detail de volledige ontwikkeling van de hardware en alle stuurprogramma’s, de inhoud van de EEPROM, de integratie met Raspbian (en nog veel meer).
- In de projectdownload is een overlay voor Raspbian (rpi-receiver-linux-rpi-4.9.y) opgenomen; dit is de speciale firmware die nodig is voor de FM-radio die in de Raspbian-kernel moet worden geïntegreerd. Een gedetailleerde installatiehandleiding legt de procedure uit.
- Om de radio naar wens te laten werken, moet Raspbian worden aangepast: de IR-afstandsbediening LIRC moet worden geïnstalleerd en de HDMI-interface moet worden omgeleid naar het LC-aanraakscherm dat op de GPIO’s is aangesloten. Dit wordt allebei uitgelegd in de installatie-instructies.
- Het toepassingsprogramma piRadio is in het Elektor-lab ontstaan. De broncode is beschikbaar zodat wijzigingen (andere layout, extra functies, aantal stationgeheugens, gebruik van de afstandsbediening van de RC5) eenvoudig kunnen worden geïmplementeerd.
- De app is ontwikkeld met de gratis C++ toolkit Qt. Meer informatie over de software-ontwikkeling (inclusief een Websocket-API) is te vinden op de Elektor Labs-pagina [6].
- Wanneer u de FM-radio met piRadio niet wilt modificeren en, voor zover het de software betreft, direct klaar wilt zijn, dan is er ook een kant-en-klare mage voor de SD-kaart van de Rasperry Pi beschikbaar.
- De radio kan ook met een andere app worden gebruikt, zoals de RDS Surveyor. Dit Java-programma zit ook samen met een installatiehandleiding in de download. De RDS Surveyor toont echt **alle** details die door het Radio Data System worden verzonden.
- De bediening van de FM-radio en de grafische gebruikersinterface is in een video te zien.
- Er is een handleiding over het gebruik van de hardware onder Volumio [7].
- Er is ook een handleiding over hoe u Qt in Raspbian invoegt [8].

Of u nu een FM-radio wilt bouwen, de hard- en/of software op bepaalde punten wilt aanpassen of de kennis die tijdens het project is vergaard voor geheel eigen ontwikkelingen wilt gebruiken — op de projectpagina is voor elk wat wils te vinden! ◀

180119-03

IN DE STORE

→ **Kale print 160520-1 V2.0**
www.elektor.nl/rpi-receiver-160520-1

→ **Raspberry Pi 3 B+**
www.elektor.nl/rpi3b+

Kristallen

Slechts twee pootjes, maar erg interessante eigenschappen!

Robert Lacoste (Frankrijk)

Het kwartskristal neemt een prominente plaats in op de huidige lijst van onderdelen die we niet kunnen missen. Maak maar eens een iets ingewikkelder apparaat open en u vindt er waarschijnlijk minstens één. Ze hebben maar twee pootjes en toch zijn we er niet zo mee bekend. Weet u bijvoorbeeld waarvoor de condensatoren dienen die ernaast zijn gemonteerd? En wat het verschil is tussen parallel- en seriekristallen?

Figuur 1: De elektronische structuur van een kwartskristal is asymmetrisch. Als er op twee zijden druk wordt uitgeoefend, bevinden de positieve ladingen zich statistisch gezien meer aan de linkerkant en de negatieve aan de rechterkant: er ontstaat een elektrische spanning tussen de twee verticale zijden.

Figuur 2: Een kristal kan worden voorgesteld door een netwerk R1/L1/C1 parallel aan een capaciteit C2. De serieresonantie is gerelateerd aan de resonantie van L1/C1, de parallelresonantie wordt veroorzaakt door C2.

Piëzo-elektriciteit

Kristallen maken gebruik van een natuurkundig verschijnsel dat piëzo-elektriciteit wordt genoemd. Dit is een eigenschap van bepaalde materialen, waaronder kwarts, op de grens van mechanica en elektriciteit. Als u druk uitoefent op twee zijden van zo'n materiaal verschijnt er een elektrische spanning tussen twee andere zijden. Omgekeerd, als u een spanning aanlegt tussen twee bepaalde zijden zal het kwarts vervormen en zich samentrekken of uitzetten. Dit verschijnsel werd in 1880 in Frankrijk ontdekt door Pierre en Jacques Curie. De eerste trouwde vervolgens met Marie Skłodowska, maar dat is een ander verhaal. Het principe van piëzo-elektriciteit zien we in **figuur 1.** Het kristal heeft een asymmetrische structuur: het kristalrooster bevat meer of minder geladen atomen met een oriëntatie die overeenkomt met de vorm van het kristal. Als er mechanische druk op het kristal wordt uitgeoefend, wordt het kristalrooster afgeplat en zullen de positieve ladingen zich statistisch gezien voor het grootste deel aan de linkerkant bevinden, terwijl er aan de rechterkant meer negatieve ladingen zullen zijn. Hierdoor ontstaat een elektrische spanning tussen de twee zijden.

Deze verschijnselen zijn microscopisch klein: verwacht niet een kristal met het blote oog te kunnen zien samentrekken, want een paar volt op een kristal veroorzaakt niet meer dan een vervorming in de orde van een micron (0,001 mm). Hoe zwak deze ook is, piëzo-elektriciteit wordt in talrijke toepassingen gebruikt voor het opwekken van geluiden (met piëzo-transducers), microbewegingen (in tunneleffect-microscopen worden piëzo-elementen gebruikt om het object te verplaatsen) of hoogspanning (in een piëzo-gasaansteker veroorzaakt een mechanische schok op een kristal een elektrische ontlading die een vonk produceert).

Van kwartskristal tot oscillator

Een tussen twee elektroden gemonteerd stukje kwarts verandert dus door de spanning op de aansluitingen lichtjes van vorm. Als er een wisselspanning wordt aangesloten, gaat het trillen. En zoals bij iedere mechanische structuur is er een bepaalde frequentie waarbij het kristal in resonantie komt. Hierbij is de stroom tussen de elektroden in fase met de spanning op de aansluitingen: het kristal gedraagt zich als een kleine elektrische weerstand R1. Hierdoor kunnen we het kristal als frequentiereferentie in een oscillator gebruiken.

Een kristal is een resonator waarvan het elektrische model bestaat uit een netwerk van drie elementen in serie: R1, L1 en C1. Condensator C1 wordt de bewegingscapaciteit van het kristal genoemd. In ieder kristal bevindt zich ook een parasitaire condensator (C2) die wordt gevormd door de capaciteit tussen de twee elektroden.
Het complete vervangschema van een kristal is dus R1/L1/C1 in serie, met C2 daaraan parallel (**figuur 2**). Als u mij vraagt waarom we een kristal gebruiken en geen echte weerstanden, condensatoren en spoelen die op dezelfde manier zijn geschakeld, dan kan ik u vertellen dat de waarden van de vergelijkbare onderdelen niet echt eenvoudig verkrijgbaar zullen zijn bij uw onderdelenhandelaar: L1 heeft gewoonlijk een waarde van meerdere henry's, terwijl de waarde van C1 wordt uitgedrukt in femtofarad (een duizendste picofarad). Door R1, die een waarde heeft van enkele honderden ohms, heeft het kristal een erg hoge kwaliteitsfactor, dat wil zeggen dat zijn resonantiegebied erg smal is, ideaal om als frequentiereferentie te dienen.

Fabrikanten van kristallen weten hoe ze een kwartskristal moeten snijden om alle waarden te optimaliseren (met standaard-sneden zoals X, Y, CT en AT). Iedere snijhoek levert een voor een bepaalde toepassing geoptimaliseerd kristal op.

Parallel- en serieresonantie

Laten we eens kijken hoe een kristal zich gedraagt als er een wisselspanning op de pennen wordt aangesloten. Als de frequentie erg laag of erg hoog is, gedraagt het netwerk R1/L1/C1 zich als een open circuit en kan het kristal worden voorgesteld door alleen condensator C2. Bij een bepaalde frequentie daar tussen in, deze kan worden berekend met de formule uit **figuur 2**, komt het netwerk R1/L1/C1 in resonantie: L1 en C1 heffen elkaar op en het kristal gedraagt zich als een weerstand (R1) met een minimale impedantie. Daarom wordt dit punt de serieresonantiefrequentie van het kristal genoemd. Deze is onafhankelijk van C2 en dus ook van parasitaire capaciteiten in de schakeling rond het kristal.
Wat gebeurt er als we de frequentie iets laten toenemen? R1/L1/C1 gaat zich nu als een spoel gedragen en komt in resonantie met C2 die hieraan parallel staat. Bij deze frequentie heeft het kristal een maximale impedantie. Dit noemt men de parallel-resonantiefrequentie. Deze hangt zeker af van de interne parasitaire capaciteiten van het kristal, maar nu hebben externe parasitaire capaciteiten ook invloed, omdat elke condensator parallel aan de contacten van het kristal de waarde van C2 beïnvloedt.

Uit de formule in figuur 2 blijkt dat de parallel-resonantiefrequentie dichter bij de serie-resonantiefrequentie komt als C2 groter wordt. Toevoegen van condensatoren rond een kristal verlaagt dus de parallelresonantiefrequentie.

Figuur 3: De meting van een kristal, uitgevoerd met een netwerkanalyzer, toont een piek van minimale impedantie bij 8 MHz, gevolgd door een piek van maximale impedantie bij een iets hogere frequentie. De andere pieken zijn parasitaire verschijnselen.

Figuur 4: De bovenste grafiek laat een verschil van 12 kHz tussen serie- en parallelresonantie zien. De toevoeging van een externe condensator van 20 pF (onderste grafiek) wijzigt de serieresonantie niet, maar brengt wel de parallelresonantiefrequentie veel dichterbij.

Omdat er niets boven een kleine demonstratie gaat, heb ik voor u een kristal van 8 MHz aangesloten op de meetpennen van mijn HP3585-spectrumanalyzer. Dit meetapparaat, dat is voorzien van een tracking-generator, injecteert een toenemende frequentie in het te testen object en laat de sterkte van het doorgelaten signaal zien als functie van de frequentie. In **figuur 3** zien we een duidelijke eerste piek waarbij de impedantie minimaal is (de piek reikt tot bovenaan het scherm, het doorgelaten signaal is sterk), bij een frequentie van 8.000.250 Hz. Deze piek wordt 12 kHz verderop gevolgd door een piek met maximale impedantie. Als we nu een condensator van 20 pF over de aansluitpunten van het kristal aanbrengen, dan ziet u in **figuur 4** dat de serieresonantiefrequentie gelijk blijft, maar dat de parallelresonantiefrequentie veel dichterbij komt en nu op een afstand van zo'n 500 Hz zit.

Dit voorbeeld toont aan dat er in principe geen verschil is tussen een kristal dat een seriekristal wordt genoemd en een kristal dat een parallelkristal wordt genoemd. In tegendeel, u kunt een kristal gebruiken in een schakeling die het in zijn serie-modus laat oscilleren (door te zoeken naar de minimale impedantie) of in zijn parallel-modus (door te zoeken naar de maximale impedantie). U zult nu ook hebben begrepen dat de frequenties van deze modi een beetje verschoven zijn. Dus als u een kristal koopt met het opschrift '8 MHz', is het handig om even te controleren voor welke modus deze frequentie geldt. Een kristal dat als seriekristal wordt verkocht zal als het als zodanig wordt gebruikt, een frequentie van precies 8 MHz hebben. En

Een onderdeel dat je absoluut beter moet leren kennen!

ook omgekeerd. Een ander aandachtspunt: zoals u hebt gezien, wordt een kristal in parallelmodus beïnvloed door externe parasitaire capaciteiten. Omdat deze nooit nul zijn, regelen de fabrikanten altijd hun kristallen op een iets hogere frequentie af, waardoor ze de juiste frequentie zullen geven als ze met een externe afstemcondensator van zo'n 10 pF worden gebruikt. De ontwerper van de schakeling kan nu de waarde van die condensator aan de andere parasitaire capaciteiten van de schakeling aanpassen. Nu weet u waarom in de meeste gevallen het kristal door een of twee kleine condensatoren wordt geflankeerd.

Figuur 5: De klassieke Pierce-oscillator, hier gerealiseerd met twee logische poorten.

Figuur 6: Met een oscilloscoop is eenvoudig te zien dat de spanningen op de aansluitingen van het kristal in tegenfase zijn (onderste grafieken). De bovenste grafiek toont de uitgangsspanning van de oscillator.

De Pierce-oscillator

We weten nu voldoende om het schema van een klassieke kristaloscillator te gaan bekijken, de oscillator van Pierce (**figuur 5**), die hier met twee logische CMOS-poorten is gerealiseerd. Nagenoeg alle CMOS-oscillatoren werken in de parallelresonantiemodus omdat een CMOS-schakeling makkelijker met een hoge impedantie werkt. C1 en C2 vormen de parallelcondensator die nodig is om de frequentie van het kristal in deze modus in te stellen. Ze zijn in serie geschakeld met hun onderlinge verbinding aan massa. Door op deze manier twee condensatoren te gebruiken in plaats van één, wordt de oscillatorwerking gestabiliseerd. Omdat ze in serie zijn geschakeld, moet hun waarde het dubbele zijn van de gewenste waarde, in dit geval dus 2 × 20 pF, dat levert 10 pF.

Een kristal dat op zijn parallelresonantiepunt zit, vertegenwoordigt een maximale impedantie en introduceert een faseverschuiving in de orde van 90°. Om een oscillator te laten werken, moet de totale faseverschuiving 360° zijn. De CMOS-poort is een inverter en levert dus een faseverschuiving die, als we rekening houden met de vertragingstijd, iets meer dan 180° bedraagt. Er ontbreken dus nog enkele tientallen graden om op 360° uit te komen. Deze worden gerealiseerd met R2, die samen met de capaciteit van het kristal een kleine vertraging introduceert. Een goed uitgangspunt voor deze weerstand is een waarde die overeenkomt met de impedantie van condensator C2 op de werkfrequentie. Deze kunnen we berekenen met $R=1/(2\pi \cdot f \cdot C)$ en dat levert in dit geval 2 kΩ op. Met weerstand R1 wordt de logische poort omgevormd tot een min of meer analoge versterker. De waarde ervan, meestal ergens tussen 100 kΩ en 1 MΩ, is niet kritisch.

Ik heb deze oscillator op een klein experimenteerbordje opgebouwd. De bijbehorende golfvormen zijn in **figuur 6** weergegeven. Zoals verwacht zijn de grafieken gemeten aan de ingang en de uitgang van het kristal dankzij weerstand R1 180° in fase verschoven en is de uitgangsfrequentie 8 MHz. In werkelijkheid heb ik 8,0021348 MHz gemeten, dat kan worden verklaard door het feit dat ik de kristaloscillator met lange draden op een experimenteerbordje heb gebouwd; hierdoor ontstaan er parasitaire serie-zelfinducties op de aansluitingen van het kristal waardoor de resonantiefrequentie wordt verhoogd. Door de waarden van C1 en C2 aan te passen kan de gewenste frequentie waarschijnlijk wat beter worden benaderd.

Opstarten...

De kristaloscillator lijkt eenvoudig, maar ervaren ontwerpers weten dat schijn bedriegt. Je kunt er bijvoorbeeld niet van uitgaan dat zo'n oscillator direct na het inschakelen van de voedingsspanning goed werkt. Een kristal is een mechanische resonator die een tamelijk complex proces doorloopt voordat de werking stabiliseert. Als u een oscilloscoop op de uitgang van een dergelijke oscillator aansluit, zult u zien dat de omhullende van het uitgangssignaal enige tijd nodig heeft om het verwachte niveau te bereiken, bij mijn experimenten was dit ongeveer 10 µs. Dit heeft te maken met de stijgtijd van de voeding en de start van het oscillatorcircuit zelf.

Als u denkt dat alles na deze 10 µs is gestabiliseerd, zult u verbaasd zijn. Om te zien wat er precies gebeurt, gebruik ik een tijd/frequentieanalyzer. Omdat dit een apparaat is dat je

niet vaak in een lab tegenkomt, is dit een mooie gelegenheid voor een demonstratie! De HP 5372A '*Frequency and time interval analyzer'* is een frequentiemeter die honderdduizenden metingen per seconde kan uitvoeren en deze in een grafiek als functie van de tijd weergeeft. We verbinden dit apparaat met de uitgang van de Pierce-oscillator en schakelen de spanning in (**figuur 7**). Aan de bovenkant van het scherm zien we dat de oscillatorfrequentie netjes 8 MHz is, maar gedurende de eerste 240 µs is de frequentie onregelmatig en loopt zelfs op tot 40 MHz!

Dit vreemde verschijnsel heeft te maken met de stabilisatie van de mechanische trillingen van het kristal en we kunnen hier weinig aan doen: iedere kristaloscillator heeft een tamelijk lange stabilisatietijd. Als u de specificaties van een microcontroller in detail bekijkt, zult u zien dat deze een interne vertraging heeft die bij configuratie voor een extern kristal de start van de kern even uitstelt.

Is hij stabiel?

Eenmaal correct opgestart is een kristaloscillator absoluut stabiel. Dat wil zeggen in relatie tot de stabiliteit die je kunt verwachten. Allereerst is er een afwijking als functie van de omgevingstemperatuur. Afhankelijk van de kwartssnede is dit effect groter of kleiner, maar niet verwaarloosbaar. U kunt een afwijking verwachten in de orde van één 'part per million' (1 ppm) per graad variatie van de omgevingstemperatuur. Voor dit probleem bestaan twee oplossingen. De eerste is de omgevingstemperatuur meten en hiermee de afwijking van het kristal compenseren. U hebt dan een TCXO (*temperature compensated crystal oscillator*) waarmee de temperatuurfout met een factor 100 kan worden verkleind. Deze compensatie kan analoog worden uitgevoerd, meestal met een varicap-diode, of digitaal. De tweede oplossing is het elimineren van temperatuurvariaties door het kristal in een verwarmde behuizing te plaatsen die op een constante temperatuur wordt gehouden. Deze oplossing, de OCXO (*oven controlled crystal oscillator*), is uiteraard wat duurder en minder energiezuinig dan de eerste, maar maakt het mogelijk om nog meer aan stabiliteit te winnen.

Is de temperatuur het enige probleem? Nee, zeker niet. Een kristal is bijvoorbeeld ook gevoelig voor mechanische trillingen: laat een schakeling met een kristal trillen met 100 Hz en u vindt vrijwel zeker 100 Hz (en de harmonischen hiervan) terug in iedere frequentie die door de schakeling wordt geproduceerd... Een ander klassiek probleem is de invloed van een slecht gestabiliseerde voeding. Hier volgt een voorbeeld: Aangezien ik toch mijn tijd/frequentie-analyzer bij de hand had, heb ik wat storing op de 5V-voeding van de oscillator aangebracht. Ik gebruikte hiervoor een laagfrequent-oscillator, ingesteld op 1 Hz en een amplitude van 50 mV, waarvan ik het uitgangssignaal via een beveiligingsweerstand van een paar ohm op de voedingslijn aansloot. Hierdoor verandert de voedingsspanning van de oscillator sinusvormig tussen 4,95 V en 5,05 V. U raadt al wat er op de uitgang te zien is? Uiteraard zien we de amplitude van het signaal variëren, maar ook, en dat is veel verrassender, de frequentie! Kijk maar eens naar de sinusoïde in **figuur 8**: de uitgangsfrequentie varieert tussen 8,002150 MHz en 8,002140 MHz, en dat is al een afwijking van 10 Hz bij een verstoring van 50 mV. Conclusie: besteed altijd zorg aan de voeding van uw oscillatoren, zelfs die met een kristal.

Figuur 7: Met een tijd/frequentie-analyzer, in dit geval een HP5372A, kan het frequentieverloop over de tijd worden weergegeven. De uitgangsfrequentie van de oscillator loopt eerst op tot 40 MHz om zich na ongeveer 240 µs op 8 MHz te stabiliseren.

Harmonische resonanties

Een kristal kan op zijn fundamentele resonantiefrequentie of op harmonische frequenties (*overtone*) worden gebruikt. Sommige kristallen zijn voor zulke toepassingen geoptimaliseerd, maar u kunt hetzelfde resultaat verkrijgen met een willekeurig kristal. We gaan nog even terug naar de kristalmeting met de net-

Figuur 8: Dit is wat er gebeurt als de voeding van de oscillator niet stabiel is: de uitgangsfrequentie is het dan ook niet!

Figuur 9: Met een kristal is vaak resonantie op een harmonische frequentie mogelijk. De piek aan de linkerkant bevindt zich bij 8 MHz en die aan de rechterkant ongeveer bij 24 MHz, ongeveer het drievoudige.

Figuur 10: Door toevoeging van een klein banddoorlaatfilter kan een kristal van 8 MHz worden veranderd in een 24MHz-oscillator.

werkanalyzer om te kijken wat er gebeurt bij veel hogere frequenties (**figuur 9**). We zien bij 8 MHz weer serieresonantie gevolgd door parallelresonantie, maar ook een andere veel kleinere resonantie bij 24 MHz. Dat is ongeveer de derde harmonische van de fundamentele frequentie, maar pas op: deze verhouding is nooit exact. Het gaat hier om een mechanische resonantie van een hogere orde die niet overeenkomt met een simpele vermenigvuldiging met drie (hoewel het er erg op lijkt). Om een oscillator op een niet-fundamentele frequentie te laten resoneren, is het in theorie voldoende om een banddoorlaatfilter in serie met het kristal op te nemen. Dit verzwakt de hoofdresonantie en forceert oscilleren op de harmonische frequentie. Zie **figuur 10**: de twee toegevoegde onderdelen (L1/C1) vormen een resonantiekring met een frequentie van 25 MHz. De uitgangsfrequentie van deze schakeling bleek bij meting 23,95966 MHz te zijn. Dichtbij, zoals voorspeld, maar niet exact gelijk aan het drievoudige van de fundamentele frequentie. Het principe klopt, maar vanwege parasitaire effecten is het iets ingewikkelder om een oscillator correct in een harmonische modus te laten werken dan je uit de theorie zou kunnen opmaken.

Tot slot

Na dit uitstapje in de wereld van de kristallen zouden de begrippen serieresonantie en parallelresonantie evenals de rol van de kleine condensatoren rond een kristal nu wat duidelijker moeten zijn voor u. Aarzel niet om nu zelf te gaan experimenteren. Informatie om u wat verder in deze materie te verdiepen vindt u in de onder weblinks genoemde artikelen. ⏮

(150013)

Dit artikel werd eerder gepubliceerd in Circuit Cellar (nr. 215, juni 2008)

Weblinks & informatie

- Fundamentals of Quartz Oscillators, Application Note 200-2, Hewlett-Packard http://leapsecond.com/hpan/an200-2.pdf
- Use of the CMOS Unbuffered Inverter in Oscillator Circuits, Application Report SZZA043 - January 2004, Texas Instruments
- www.ti.com/lit/an/szza043/szza043.pdf
- Piezoelectricity Encyclopedia Universalis
- HCMOS Crystal Oscillators, Fairchild Semiconductor Application Note 340, May 1983, www.fairchildsemi.com/an/AN/AN-340.pdf
- Oscillators for Microcontrollers, AP-155 application note, Intel, http://ecee.colorado.edu/~mcclurel/iap155.pdf
- http://en.wikipedia.org/wiki/Crystal_oscillator
- www.doveonline.com/oscillator/more-on-crystal-oscillators.php

Tips en trucs

Van lezers voor lezers

Hier is weer eens zo'n slimme oplossing die het leven van elektronici eenvoudiger maakt.

Digitale camera als testbeeldgenerator

Alfred Rosenkränzer (alfred_rosenkraenzer@gmx.de)

Om een video-bewakingscamera in te regelen, had ik een tijdje terug een kleine, mobiele en batterijgevoede video-testbeeldgenerator nodig. Uit mijn tijd als professioneel ontwikkelaar van videotechniek bezit ik weliswaar nog diverse generatoren, maar die zijn noch klein, noch batterijgevoed. Daarom begon ik na te denken over wat er met hedendaagse spullen als CPLD, RAM en DAC maakbaar is.

Figuur 1. Het testbeeld op mijn drie camera's.

Tot ik me realiseerde dat ik (en vermoedelijk u ook) zoiets al bezit: dat zit namelijk allemaal in een digitale camera. Die kan niet alleen de gemaakte foto's op het eigen display weergeven, maar ook op een externe monitor. Oudere camera's hebben daartoe 'slechts' een analoge uitgang (vaak een 3,5 mm stekerbus), nieuwere vaak ook een HDMI-uitgang.

Testbeelden zijn op internet genoeg te vinden, bijvoorbeeld bij de firma Burosch [1]. Maar helaas wilde geen van mijn camera's een foto weergeven die hij (of een van zijn broertjes) niet zelf had gemaakt. Mijn pogingen om in de bits en bytes van het EXIF-bestand de juiste te modificeren locatie te vinden, heb ik na enige tijd maar opgegeven. Mocht een lezer wel weten hoe dat moet, dan houd ik me van harte aanbevolen.

Figuur 2. Het zelf gefabriceerde multiburst-testbeeld.

Om dit probleem te omzeilen, heb ik met de camera simpelweg een foto gemaakt en vervolgens met een beeldbewerkingsprogramma de beeldinhoud ervan vervangen. Voor de analoge uitgang volstaat een lage resolutie van 640 x 480 of 800 x 600 pixels, meer wordt sowieso niet weergegeven. De videonorm voor PAL is zoals bekend 720 x 576 en voor NTSC 720 x 480 pixels.

***Figuur 1** toont het zo gemaakte testbeeld op mijn drie camera's. Om de frequentiekarakteristiek van de transmissieweg (bij mij zo'n 80 m) te bepalen, kan een speciaal videotestsignaal worden gebruikt dat multiburst wordt genoemd. Zoals de benaming al aangeeft, bestaat dat uit meerdere bursts, beginnend met een lage frequentie en doorlopend tot de bovenste grensfrequentie van het systeem (ca. 5 MHz bij PAL en 4,2 MHz bij NTSC). Een multiburst was ook onderdeel van het genormeerde testbeeld, dat vroeger in de pauzes tussen TV-uitzendingen werd uitgezonden. Aan het eind van de transmissieweg kan het testbeeld op een monitor of het ontvengen signaal op een oscilloscoop worden bekeken. Als alle frequentiebursts dezelfde amplitude hebben, dan is de frequentiekarakteristiek vlak en de bandbreedte groot. De bursts kunnen in één lijn worden weergegeven of van boven naar beneden verdeeld over het beeld. Men spreekt dan van een horizontale resp. verticale multiburst.*

*Omdat ik een dergelijk testbeeld niet kon vinden, heb ik het zelf gemaakt (geïnteresseerde lezers kunnen per e-mail bij mij de details opvragen). Aan de bovenste en onderste rand worden het maximale (wit) en minimale (zwart) signaalniveau als referentielijnen weergegeven. Het begin en eind van elke lijn verloopt vloeiend, om storende uitslingering te voorkomen. Het resultaat is in **figuur 2** te bewonderen.*

De D/A-converter en het daaropvolgende filter van de camera hebben helaas niet de kwaliteit van een professionele testbeeldgenerator. Ook de samplefrequentie is niet altijd groot genoeg voor het gehele 5-MHz-spectrum. Daarom zullen we enige beperkingen in de maximale signaalfrequentie en beeldkwaliteit moeten aanvaarden. De door mij gebruikte beeldbestanden zijn te downloaden van de bij dit artikel horende Elektor-webpagina [2]. Als u net zo'n weerspannige camera hebt als ik, kunt u zoals hiervoor uiteengezet de inhoud van een zelfgeschoten plaatje met behulp van Photoshop & Co. vervangen door een testbeeld. ◀

(160426)

[1] www.burosch.de/tv-bildeinstellungen/testbilder.html

[2] www.elektormagazine.nl/160426

Heb je zelf iets slims of handigs bedacht ? Een onderdeel of gereedschap op ongebruikelijke wijze toegepast? Of heb je een idee hoe je een probleem anders kunt benaderen dan tot nu toe gebruikelijk? Schrijf ons – wij belonen elke tip die wij publiceren met €40!

Verilog Basics

beginselen van de hardware-beschrijvingstaal Verilog

Jörg Zollmann (Duitsland)

Verilog is naast VHDL de bekendste en meest verbreide hardware-beschrijvingstaal (HDL) voor CPLD's en FPGA's. In dit artikel geven we u een overzicht van de belangrijkste elementen van deze taal.

Even ter herinnering: met een hardware-beschrijvingstaal kan een digitale schakeling in tekstvorm worden gespecificeerd. Vervolgens kan de schakeling met behulp van simulatie functioneel worden getest. Een andere functie van de HDL is het omzetten van de beschrijving in echte hardware (synthese) met behulp van software-tools.

Verilog en VHDL hebben beide zowel in de industrie als onder hobbyisten een grote gebruikersgroep. Voor iemand die aan een CPLD- of FPGA-project werkt kan het dus een voordeel zijn om de basisprincipes van beide talen te kennen. Met de in deze handleiding beschreven basisprincipes van Verilog zult u al veel projecten zelf kunnen realiseren en grotere ontwerpen, zoals de in de komende ElektorLabs-edities te presenteren 'DIY-processor', kunnen begrijpen.

Conventies

De Verilog-syntax lijkt sterk op die van C. De leercurve is bij deze taal in het algemeen niet zo steil als bij VHDL, en er kunnen al snel en met aanzienlijk minder regels code goede resultaten worden bereikt. In tegenstelling tot VHDL maakt Verilog wel onderscheid tussen hoofd- en kleine letters. De broncodebestanden hebben standaard de vorm *.v (*.vh , *.sv, *.svh). Commentaar wordt in Verilog voorafgegaan door een dubbele slash `//`. Commentaar dat uit meerdere regels bestaat wordt tussen `/*` en `*/` geplaatst. Spaties worden door Verilog genegeerd. Toegestane *identifiers* (aanduidingen van variabelen) zijn alfanumerieke tekens, de underscore en het dollarteken (`a-z A-Z 0-9 _ $`). Het dollarteken mag echter niet aan het begin van een naam staan. Het equivalent van accolades in C (`{,}`) is in Verilog `begin` en `end`. Net als bij software-ontwikkeling zijn ook bij Verilog verschillende codeerstijlen ontstaan, waarbij de inspringdiepte en het aantal begin/end-statements van elkaar verschillen. Het kan voor zowel beginnende als gevorderde gebruikers van deze taal handig zijn om elk sequentieel blok en iedere voorwaardelijke vertakking van een `begin`/`end` te voorzien.

Getalrepresentatie en datatypen

Getallen worden in Verilog gedefinieerd volgens de regel `<size>'<signed><radix> number`, waarin `<size>` het benodigde aantal bits weergeeft en niet het aantal gebruikte tekens. De vermeldingen tussen vierkante haken zijn optioneel, de standaardinterpretatie van getallen is *unsigned decimal*. Het getal 42 kan dus bijvoorbeeld als volgt worden weergegeven:

Tabel 1. Vierwaardige logica in Verilog.

Logische waarde	Interpretatie/evaluatie
0	Logische 0 (false)
1	Logische 1 (true)
x	Onbekend (don't care)
z	Hoogohmig

```
6'b101010  // binair (MSB aan de kant van de ' )
6'o52'     // octaal
6'h2a      // hexadecimaal
6'd42      // decimaal
6'b10_1010 // De underscore '_'  wordt gebruikt om de
   leesbaarheid te verbeteren
'd42       // zonder prefix worden er 32 bits voor
   het getal gebruikt
```

Verilog kent slechts twee datatypen. Het ene is het Net-type, en de voor de synthese belangrijkste vertegenwoordiger hiervan is `wire`. Zoals de naam al aangeeft wordt een wire gebruikt om verbindingen te modelleren. Het andere datatype is het Register-type waarvan `reg` de belangrijkste vertegenwoordiger is. Integer, real, time en realtime zijn ook Register-typen, maar deze zijn voor de synthese van minder belang. Het Register-type is een geheugentype. Tussentijdse opslag van waarden moet dus met een `reg` worden gedaan. Dit betekent echter niet automatisch dat bij het toepassen van een `reg` ook altijd een echt geheugenelement (flipflop) wordt gesynthetiseerd (later meer hierover). De twee datatypen van Verilog bevatten vierwaardige logica (zie **tabel 1**). Voor beide datatypen zijn rekenkundige en logische operaties gedefinieerd. In **Tabel 2** vindt u de belangrijkste operatoren met hun functie en een klein voorbeeld.

Als er geen type wordt gedeclareerd, neemt Verilog aan dat een `wire` (1 bit) wordt bedoeld. Als u de typedeclaratie van een signaal weglaat zal het synthese- of simulatietool geen foutmelding geven, maar alleen een waarschuwing dat de onbekende variabele automatisch als 1-bit-wire wordt beschouwd. In Verilog hoeven geen 'packages' te worden toegevoegd of functies voor typeconversie worden opgeroepen. Dit maakt de code simpeler en compacter, maar heeft als nadeel dat een Verilog-compiler vaak dingen toelaat die in VHDL niet mogelijk

zouden zijn. Zo leidt bijvoorbeeld de toewijzing van vectoren met verschillende breedte vaak alleen tot een *Warning (truncation)*. Het gevolg is dat de gewenste schakeling eventueel niet synthetiseerbaar is, of een andere functie heeft dan u zou verwachten. Als voorbeeld declareren we een vector van het type `reg [7:0]`, een 8-bits breed register, en wijzen hier verschillende waarden aan toe:

```
reg [7:0] bus1 = 8'b1011; /* bus1 = 00001011 MSB's
   worden met 0 gevuld */
reg [7:0] bus2 = 3'b1101; /* bus2 = 00000101 Bit 4
   wordt afgekapt (truncated) */
```

Bij de evaluatie van de waarde van de getallen heeft de size-aanduiding voorrang. Bij de hierop volgende toewijzing van de waarde aan de reg [7:0] bus worden de MSB's met nullen gevuld.

Module

De basiseenheid van de code is bij Verilog de module. Een Verilog-module begint met het keyword `module` en eindigt met `endmodule`. Na het keyword volgt de naam van de module, dan een optionele parameterlijst en vervolgens de poortlijst. De poortlijst bevat de namen van alle in- en uitgangen. In eerdere versies van Verilog werden alleen de signaalnamen vermeld, maar sinds 2001 is het toegestaan om ook de poortdeclaratie, oftewel de signaalrichting (ingang/uitgang) evenals de type-declaratie van de signalen in de poortlijst op te nemen. Na de poortlijst volgt de eigenlijke functionele beschrijving van de module met verschillende toewijzingen en statements.

```
module a (input  wire  a);
// Lege module
endmodule
```

Verilog ondersteunt een hiërarchisch schakelingontwerp. Een module kan dus extra submodules instantiëren. Als deze een ander gedrag moeten vertonen dan de oorspronkelijke module, kan dit met een parameter worden ingesteld. Een `parameter` kan door de parameterlijst aan een module worden overgedragen, deze lijst onderscheidt zich van de poortlijst door het teken #.
Hier volgt een voorbeeld van een module met een parameter `N` met defaultwaarde 1. Deze waarde kan bij het instantiëren van module `m` worden gewijzigd. Module `m` bestaat uit een submodule `a_1`, die een instantie van de eerder genoemde module is. Bij het instantiëren van modules worden de signalen van de bovenliggende instantie aan de onderliggende doorgegeven, hetzij expliciet met de signaalnamen of impliciet via de positie in de poortlijst (zie **listing 1**).

Tabel 2. Operatoren in Verilog.

Logische operatoren	**Beschrijving**	**Voorbeeld** met :reg [1:0] A = 2'b01;reg [1:0] B = 2'b10;reg [1:0] C = 2'b00;
!	Logische NOT	! A; // logische 0
&&	Logische AND	A&&B; // „1 && 1" = logische 1A&&C; // „1 && 0" = logische 0
||	Logische OR	A||C; // „1 || 0" = logische 1
Bit-operatoren		
~	Not	C&(~A);
&	And	A&B;
|	Or	A|C;
^	Xor	A^B;
Rekenkundige operatoren		
*	Vermenigvuldigen	C= A*B // = 2'b10
/	Delen	C= A/B // =2'b00
+	Optellen	C = A+B // = 2'b11
-	Aftrekken	C = B-A // = 2'b01
**	Exponent	
%	Modulo	
Vergelijkende operatoren		
>; <;	groter; kleiner;	if (A>B)||(A>C)
>=; <=;	groter of gelijk; kleiner of gelijk;	if (A>=B)||(A>=C)
==	gelijk aan	if (A == B)
Overige operatoren		
>>	Schuiven naar rechts	A<<1 // A=2'b10
<<	Schuiven naar links	B>>1 // B=2'b01
{}	Samenvoegen	{A,B} // 4'b0110
{{}}	Replicatie	{2{A}} // 4'b0101

```
module m #(parameter N = 1) ( input wire a);
// Submodule a: Instance a_1
a a_1(.a(a)); // Van module a wordt instantie a_1
   gecreëerd
// Als de geïnstantieerde module a ook een parameter
   N zou hebben
// zou de regel er zo uitzien: a #(.N(10))a_1(.a(a));
endmodule
```

Steeds opnieuw

Het `always`-statement is in Verilog een belangrijk keyword. Hiermee wordt een procedure-blok aangeduid dat, zoals de naam al aangeeft, steeds weer wordt aangeroepen. Bij de synthese wordt het gebruikt voor de beschrijving van combinatorische en sequentiële logica. Aan een `reg`-type mag alleen binnen een always-blok een signaal worden toegewezen. Een always-blok kan een *sensitivity*-lijst hebben die ervoor zorgt dat de in het blok gedefinieerde *statements* alleen dan worden geëvalueerd als na het meest recente simulatietijdstip een waarde is toegewezen aan een signaal in de lijst. Hierdoor is het always-blok vergelijkbaar met een VHDL-proces. De sensitivity-lijst begint in Verilog met het `@`-teken. Alle signalen die tussen de haken na dit teken als onderdeel van een or-configuratie worden opgevoerd, worden gemonitord. Als slechts een van de flanken van een signaal moet worden gemonitord, wordt dit met het keyword `posedge` of `negedge` aangegeven.
Voor de beschrijving van een sequentiële synchrone schakeling wordt `always @ (posedge clk)` gebruikt. Dit statement zorgt ervoor dat alle blokken van een ontwerp, ook van verschillende hiërarchische niveaus, die met hetzelfde kloksignaal (`clk`) zijn verbonden op hetzelfde moment worden uitgevoerd. Dit vormt voor beginners vaak het grootste mentale struikel-

Listing 1. Logische 'cloud'

```
module logic_cloud_verilog
(
  input  wire  a,
  input  wire  b,
  input  wire  c,
  output reg   d,
  output reg   e
);

wire [3:0] abc;

// continuous assignment
// Left hand side operator of continuous assigments must
be
// wire-type
assign  abc = {a,b,c};  // concatenation of a,b c and d
                        // with {}

// the * in the sensitivity list is a wildcard
// this means every signal which is in the block
// triggers the block
always @ (*) begin
  d = 1'b0; // This default assignment helps to avoid
            // inferred latches
  if (abc == 3'b001) begin // begin..end is optional
    d = 1'b1;
  end else if (abc == 3'b101) begin
    d = 1'b1;
  end
end

always @ (*) begin // begin..end is optional
  case (abc)
    3'b000  : e = 1'b1;
    3'b011  : e = 1'b1;
    default : e = 1'b0;
  endcase
end

endmodule
```

Listing 2. D-FF met asynchrone reset en klokvrijgave

```
module d_ff_verilog
(
 input  wire clock,
 input  wire reset_n,
 input  wire ena,
 input  wire d,
 output reg  r_q,
 output reg  r_qn,
 output wire w_qn,
 output wire w_notOut

);

assign w_qn = ~r_q;      // continuous assignment -> this
                         // will be a not buffer
not not_1(w_notOut,r_q); // usage of Verilog Primitve

// Verilog-2001: permits comma-separation in
// sensitivity list
// always @ (posedge clock, negedge reset_n)
always @ (posedge clock or negedge reset_n) begin
  if (reset_n == 1'b0) begin
    r_q  <= 1'b0;          // reset value of output
    r_qn <= 1'b1;
  end else begin
    if (ena == 1'b1) begin // clock enable
      r_q  <= d;          // continuous assignment -> this
                          // will be a not buffer
      r_qn <= ~r_q;       // this translates to a second FF
                          //  with inverting output
    end
  end
end
endmodule
```

blok. Binnen het always-blok bevinden zich procedurele taken (*Assignment*), die na elkaar worden uitgevoerd. Er zijn twee soorten assignments: de *Blocking Assignments* (A = B;) en de *Non-Blocking Assignments* (A <= B;). Bij Blocking Assignments (=) wordt de evaluatie van de taak eerst volledig uitgevoerd voordat met de volgende wordt begonnen. Bij Non-Blocking Assignments (<=) worden al opdrachten uitgevoerd voordat de voorgaande gebeurtenissen volledig zijn geëvalueerd. Wanneer een bepaald type assignment moet worden gebruikt is niet altijd duidelijk, en dit kan niet worden vastgesteld zonder diepgaande kennis van de interne structuur van de Verilog-simulator, die een onderdeel van de taalspecificatie vormt. Critici van Verilog beschouwen dit vaak als de belangrijkste zwakke plek van de taal. In de praktijk speelt dit echter een ondergeschikte rol, want er is een vuistregel die luidt: als combinatorische logica moet worden beschreven gebruik je een Blocking Assignment (=). In het geval van sequentiële logica gebruik je een Non-Blocking Assigment (<=). Wie zich aan deze regel houdt zal afgezien van een paar exotische uitzonderingen geen problemen ondervinden. Wilt u toch weten hoe het precies zit met deze verschillende assignments, dan kunt u daar bijvoorbeeld in [1] meer informatie over vinden.

Voorbeeld

Listing 1 geeft een voorbeeld van een module waarin uitsluitend combinatorische logica wordt beschreven. Volgens de hierboven vermelde vuistregel zijn alle assignments in deze module dus Blocking-Assignments (=). De schakeling heeft niet echt een zinvolle functie, maar laat een aantal taalconstructies zien die de moeite waard zijn om nauwkeuriger te bekijken. Om te beginnen wordt een 3-bits brede vector van het type `wire` gedeclareerd. Vervolgens worden aan deze `wire` via een `continuous`-assignment de waarden van de ingangen a,b,c toegewezen. Omdat a, b en c 1-bit-wires zijn worden ze voor de toewijzing aan een 3-bit wire samengevoegd met de concatenation-operator `{}`. **Let op:** Aan de linkerkant van een continuous-assignment moet een wire-type staan. In het eerstvolgende always-blok wordt het gebruik van `if/else` geïllustreerd. Een vuistregel die voorkomt dat per ongeluk een latch wordt gecreëerd, schrijft voor dat alle variabelen die binnen dit blok een toewijzing kunnen krijgen, deze ook onder alle omstandigheden echt moeten krijgen. Een eenvoudig alternatief hiervoor is om aan het begin van het blok alle variabelen een onconditionele default-toewijzing te geven; zoals de toewijzing `d = 1'b0;` laat zien. In het tweede always-blok is een voorbeeld van een `case`-statement te zien. Dit werkt op dezelfde manier als in C. De `default`-toewijzing aan het einde van het case-statement is weliswaar optioneel, maar moet om problemen te voorkomen altijd plaatsvinden.

Listing 2 bevat de beschrijving van een flipflop met asynchrone reset en klokvrijgave. Volgens de eerder vermelde vuistregel zijn alle assignments die sequentiele logica beschrijven Non-Blocking Assigments (<=). In dit voorbeeld zijn dat alle toewijzingen binnen het always-blok, waarin een inverterende en een niet-inverterende flipflop worden beschreven. De niet-inverterende uitgang van de flipflop wordt bovendien

Advertentie

Figuur 1. D-flipflop in de RTL-Viewer.

aan het begin van de module met behulp van een *primitive* geïnverteerd. (Base-)primitives zijn in Verilog voorgedefinieerde modules (bijvoorbeeld NOT, AND, OR, XOR), die net als 'gewone' modules kunnen worden geïnstantieerd. Hier wordt de module `not_1` geïnstantieerd waarvan de uitgang aan `wire w_notOut` wordt toegewezen en de ingang aan reg r_q. De toewijzing `assign w_qn = ~r_q;` een regel verder naar boven heeft het zelfde effect als de invertering met de base-primitive. In **figuur 1** ziet u hoe het synthese-tool Quartus de beschrijving uit listing 2 vertaalt. Zoals verwacht worden de twee uitgangen `w_not_Out` en `w_qn` door inverterende buffers aangestuurd en zijn de reg-uitgangen direct met de uitgangen van de flipflops verbonden. In de figuur is ook te zien dat de twee signalen uit de sensitivity-list op de twee flankgevoelige ingangen van de flipflops zijn aangesloten.

Simulatie

Voor de simulatie wordt in Verilog een module gebruikt die de schakeling die getest moet worden als Device-Under-Test (DUT) instantieert en met de juiste signalen aanstuurt. Deze module wordt met Testbench aangeduid. In **Listing 3** vindt u een Testbench voor de D-flipflop uit listing 2. In de Testbench, die in het algemeen niet synthetiseerbaar is, wordt in een always-blok een kloksignaal opgewekt. Om signalen in de tijd te vertragen wordt in Verilog het teken `#` gebruikt. Het in de Testbench gebruikte `initial`-blok bezit dezelfde eigenschappen als een always-blok, met als verschil dat het slechts één keer aan het begin van de simulatie wordt uitgevoerd. Om een text-string naar de simulator-uitgang te sturen wordt de systeemfunctie `$display` gebruikt (Verilog-systeemfuncties zijn niet synthetiseerbaar en beginnen met een $).

Conclusie

Verilog beschikt natuurlijk over nog veel meer functies die in dit korte artikel niet allemaal aan de orde zijn gekomen, maar de hier gepresenteerde voorbeelden vormen een goede basis van waaruit u veilig de Verilog-wereld kunt verkennen. Wie geïnteresseerd is in nog meer Verilog, kan met behulp van Eda-Playground [2] een reeks Verilog- en VHDL-voorbeelden doorwerken of een van de vele online-tutorials volgen [3].

(180562-03)

Listing 3. Testbench D-FF

```
'timescale 1ns/1ps
// timescale is a compiler directive this means
// 1 time step equals 1 ns and the resolution is 1 ps
module tb_dff;

reg  clk;
reg  rst_n;
reg  r_d;
wire w_q;
wire w_qn;
wire w_qn_1;
wire w_qn_2;

// create 50 MHz clock
always begin
  #10; // delay for 10 time steps = 10 ns before
       // inverting clk_tb again
  clk = ~clk;
end

d_ff_verilog DUT(
     .clock    (clk),
     .reset_n  (rst_n),
     .ena      (1'b1),
     .d        (r_d),
     .r_q      (w_q),
     .r_qn     (w_qn),
     .w_qn     (w_qn_1),
     .w_notOut (w_qn_2)
     );

initial begin
  $display("Hello World ");
  clk   = 0;
  rst_n = 0;
  r_d   = 1'b0;
  #100;
  r_d = 1;
  rst_n = 1;
  #200;
  $stop();
end

endmodule
```

Weblinks

[1] www.sunburst-design.com/papers/Cummings-SNUG2000SJ_NBA.pdf

[2] www.edaplayground.com/

[3] www.nandland.com/verilog/tutorials/index.html

SHOP-HIGHLIGHT

Waar elektronici van dromen…

…en hoe dromen werkelijkheid worden in de Elektor-shop

Aan de slag met FPGA's…

Voor de meeste elektronici is de greep naar een microcontroller de gewoonste zaak van de wereld wanneer er iets moet worden gebouwd dat ingewikkelder is dan een knipper-LED. En inderdaad — schakelingen waarvoor vroeger een hele zak vol TTL- of CMOS-IC's nodig was, kunnen nu in een enkel chipje worden gerealiseerd.

In veel gevallen zou een FPGA (*Field Programmable Gate Array*) een prima alternatief zijn voor een microcontroller, maar FPGA's, die inmiddels al weer zo'n dertig jaar op de markt zijn, hebben enkele zwaarwegende nadelen: de complexiteit en de kosten. Dankzij TinyFPGA kan dat gaan veranderen. Dit bedrijf levert kleine, goedkope en gemakkelijk te gebruiken FPGA-breakout-boards. Voor het ontwikkelen van FPGA-toepassingen propageert TinyFPGA gratis open source-software. De FPGA's die TinyFPGA gebruikt, stammen allemaal van chipsbakker Lattice Semiconductor. Momenteel zijn vier breakout-boards leverbaar, met verschillende FPGA-chips. Een configuratie (een 'programma') wordt in de chip geladen in de vorm van een bitstream; indien de chip geen intern niet-vluchtig geheugen bezit, moet die bitstream telkens bij het inschakelen opnieuw worden ingelezen — wat het risico van hacken met zich meebrengt, iets waaraan bedrijven een grote hekel hebben. Die bitstream-formaten worden dan ook angstvallig geheim gehouden. Niettemin is de bitstream voor de Lattice iCE-40 'gedocumenteerd' (lees: gehackt...) zodat de chip met open source-ontwikkeltools kan worden gebruikt, in dit specifieke geval Apio. Dit is een commandline-tool, maar in het huidige tijdperk van grafische interfaces is er ook een Apio-IDE, die op Atom is gebaseerd, en de (experimentele) visuele hardware-editor Icestudio waarmee schema-invoer mogelijk is.

We willen niet beweren dat FPGA-programmeren nu kinderspel is, maar met TinyFPGA en Apio komt het binnen handbereik van veel meer (hobby-)elektronici.

…en warmtebeeldcamera's

Met de toenemende miniaturisering van elektronische schakelingen en de bijbehorende voedingsschakelingen wordt het steeds belangrijker de warmteontwikkeling in de gaten te houden — maar dat is gemakkelijker gezegd dan gedaan. In professionele laboratoria worden daarvoor al lange tijd warmtebeeldcamera's gebruikt, maar die apparaten hebben een groot nadeel: er hangt een professioneel prijskaartje aan — hobbyisten kunnen er alleen maar van dromen.

Maar met de HT-02 warmtebeeldcamera is dat voorbij. Deze camera, die in China wordt geproduceerd, biedt eigenlijk alle mogelijkheden die zijn duurdere broertjes en zusjes (bijvoorbeeld die van Flir of Fluke) ook hebben, maar dan voor een aanzienlijk vriendelijker prijs. Hoe dat mogelijk is? Voornamelijk door genoegen te nemen met een geringere resolutie.

Net als bij duurdere exemplaren biedt de HT-02 de mogelijkheid een IR-beeld op een 'gewoon' beeld te superponeren om hotspots gemakkelijker te kunnen identificeren, maar bij het 2,4-inch-beeldscherm van de camera stoort die geringere resolutie eigenlijk helemaal niet.

Het temperatuurbereik van de camera met 300 °C is behoorlijk hoog en de nauwkeurigheid is met ±2 °C voor normaal gebruik groot genoeg. Met een gewicht van niet meer dan 320 g ligt de camera prettig in de hand.

De HT-02 wordt met een goede handleiding en een stevige hoes geleverd. ⏮

180689-0

IN DE STORE

→ TinyFPGA BX
www.elektor.nl/tinyfpga-bx

IN DE STORE

→ Thermal Imaging Camera HT-02
www.elektor.nl/handheld-thermal-imaging-camera-ht-02

Netspanning filteren en stabiliseren, zo deed men dat vroeger

Hoe houd ik mijn zekering heel

Jan Buiting, Beheerder/Redacteur, Elektor Retrotronica

Omdat het tegenwoordig zo stabiel en betrouwbaar is, gaat men er te vaak van uit dat de netspanning op de stopcontacten in onze elektronische werkruimten dik in orde is. Pas als je een schok krijgt, er een zekering uitknalt of de apparatuur zich grillig begint te gedragen, vermoeden we dat er problemen zijn aan de 230V-zijde. Dat is totaal anders als je werkt met vintage elektronische apparatuur, vooral als je kostbare spullen voor de eerste keer ingeschakelt ... dan is “de netspanning” van het grootste en zelfs dodelijk belang. In dit artikel laten we enkele opties zien waarmee we voorkomen dat onze lievelingen er aan gaan nog voordat ze wakker worden na jarenlang sluimeren en verwaarloosd te zijn. En laten we onszelf daarbij ook niet vergeten.

Figuur 1. De Elektor Labs 850VA-variac is nu toevertrouwd aan de *Fingerspitzen* van Retrotronica.

Welkom bij wederom een aflevering van Retrotronica, de langst lopende serie artikelen in de geschiedenis van ons tijdschrift Elektor. Deze keer iets over de netspanning waarmee we (voorzichtig) vintage apparatuur voeden en over het veilig distribueren ervan in de e-workspace of het lab. Welke opties staan er eigenlijk tot onze beschikking?

Variac

Variac was een Amerikaans handelsmerk van General Radio voor hun “variabele autotransformator”-producten. Een variac biedt een betrouwbare manier om de netwisselspanning te varieren van ongeveer 10 VAC tot iets boven de “nominale waarde” (250 VAC of zo). De basisvariac wordt bediend via een grote knop met een schaalverdeling die de uitgangsspanning bij benadering aangeeft. Hiermee verplaats je een loper van grafiet naar een positie tussen de uiteinden van de spoelwikkeling van de autotransformator die is gemaakt van koperdraad. De relatieve positie van de schuifregelaar bepaalt de uitgangsspanning, zoals een loper in een potentiometer de weerstand bepaalt.
Sommige grote variacs hebben motorbesturing. Deze apparaten waren meestal bedoeld voor afstandsbediening, met als duidelijk voordeel dat je geen omvangrijk, extreem zwaar en eeuwig brommend ding op of onder je werkplek hebt.

Variacs hebben inducties van tientallen Henries en kunnen bij inschakelen een vreemd geluid produceren (bijv. een korte fluittoon), zelfs zonder belasting. Of ze blazen de huiszekering op. Een variac heeft slechts één wikkeling die, hoewel "afgetakt", de belasting niet van het net isoleert. Het is bij beginners een veel voorkomende misvatting dat de lage 10 of 40 volt AC of zo van een variac een veilige spanning betekent. 'Laag' betekent hier zeker niet 'veilig': de volledige spanning van de fase (L) kan op een aansluiting aanwezig zijn, ervan uitgaande dat het stopcontact niet gepolariseerd is (zoals in Nederland).
De variac van **figuur 1** is al jaren een trouwe metgezel in Elektor Labs, maar werd vanwege het weinige gebruik "vergeten". Het is een 4-amp type, goed voor zo'n 850 VA. Ik herinner me dat hij veelvuldig werd gebruikt bij Elektors zware eindversterkerprojecten, als hulpmiddel om de voedingen voorzichtig op spanning te brengen en opgeblazen zekeringen en algemene chaos in het lab te vermijden. Tegenwoordig wordt hij weer dankbaar gebruikt bij het voorzichtig inschakelen van 'nieuwe' Retrotronica-apparatuur — meestal op buizen gebaseerde apparatuur met het inherente gevaar van bloemkolende hoogspanningselko's. Oorspronkelijk is het een 220-volt type uit de 70-er jaren, de schaal wijkt dus iets af van de huidige 230 VAC. De voltmeter erop is niet echt betrouwbaar, boven de 150 VAC of zo gebruik ik een Fluke DVM. Verder is de variac van onschatbare waarde voor mij en de eerste "tool" om erbij te nemen wanneer er verse vintage-apparatuur is aangekomen.
Voor de elektrische veiligheid en om indruk te maken is de Elektor Retrotronica-variac ingebouwd in een behuizing van plexiglas (acrylplaat) (**figuren 2** en **3**) met een draaggreep van PVC elektriciteitsbuis! Vooral jongeren vinden hem zwaar met zijn ongeveer 12 kilogram en de loper vonkt een beetje als hij de volle 800 watt voor zijn kiezen krijgt.
Vooral bij schakelende voedingen wordt een variac niet aanbevolen. Ik heb ooit een oscilloscoop-voeding van het type omvormer/oscillator beschadigd door de variac te gebruiken om de voedingsspanning langzaam op te voeren. Te langzaam, omdat de voeding het nominale niveau binnen een bepaalde periode verwacht. Ver onder dit niveau (bv. 150 van de 220 V), begint de zaak niet te oscilleren en loopt er overmatig en langdurig gelijkstroom door bepaalde transformatorwikkelingen met soms fatale gevolgen. Een variac wordt ook afgeraden bij sterk inductieve of capacitieve belastingen, omdat resonantie kan optreden. Variacs vind je tegenwoordig voor belachelijk lage prijzen, want de meeste verzamelen alleen maar stof of fungeren als deurstopper. Ik zou zeggen, grijp je kans.

Stabilisatie van de netspanning

Vroeger, in gebouwen waar zware elektrische machines en "kantoorelektronica" vrolijk werden gevoed vanuit dezelfde dikke voedingskabel, waren spanningspieken, -dips, uitval en enorme storing een constante bron van problemen voor ... de kantoormedewerkers. Sommige leveranciers van telefoonapparatuur weigerden zelfs om centrales en andere "gevoelige" apparatuur te installeren als de netvoeding zwaar vervuild bleek te zijn. Hetzelfde geldt voor kantoorverlichting, rekenmachines en een enkele computer of server. Denk ook eens de extra spanningsverliezen in de VS, met hun 110-117 VAC in plaats van 220 VAC. In laboratoria en fabrieken werden daarom netstabilisatoren geïnstalleerd op zorgvuldig gekozen punten, zowel elektrisch als akoestisch! Zo'n stabilisator zie je in **figuur 4**, in bedrijf en wel binnen de specificatie zoals vastgesteld door de Fluke

Figuur 2. Achterkant van de 'draagbare' variac in de transparante behuizing van plexiglas. De schuifregelaar ('loper') is niet zichtbaar; die zit tussen de spoel en de ronde afdekplaat.

Figuur 3. Er zit een aardige hoeveelheid kostbaar koperdraad in een variac. Hier zie je de schroefklemplaat en de autotransformator windingen met nog wat stof op de wikkeling (sorry).

Figuur 4. Dertig jaar of meer verwaarlozing heeft de passief onderhouden stabiliteit van de PE1033 niet beïnvloed: de uitgangsspanning is nog steeds stabiel binnen 1% van de opgegeven 220 VAC. Ingangsspanning: 232 VAC.

Figuur 5. De PE1033 bestaat uit niets anders dan kolossale transformatoren en condensatoren.

Figuur 6. Iemand heeft een porseleinen stopcontact aan de PE1033-behuizing bevestigd. Dat suggereert draagbaarheid van het instrument, wat zeker niet het geval is, het ding is bedoeld voor stationair gebruik.

Figuur 7. De Philips PE1033-netstabilisator uit ongeveer 1960 was ontworpen voor permanente aansluiting in 220 VAC elektrische systemen in laboratoria, fabrieken en kantoren, en installatie op een vaste plaats.

DVM-waarde van 218,4 VAC! Het grijze ding is een Philips PE1033 uit het midden van de jaren zestig, met kenmerken:

- Uitgangsspanning: 220 VAC ±1%
- Uitgangsvermogen: 250 VA
- Ingangsspanning: 187–242 VAC
- Ingangsvermogen: 315–335 VA
- Frequentie: 50 Hz

Behalve de informatie van het label op het apparaat, heb ik niets over dit apparaat (schema's e.d.) kunnen vinden. Binnenin zitten alleen maar vier transformatoren, drie hoge condensatoren (5 µF / 500 V) en de bedrading, dat is alles (**figuur 5**). De PE1033 weegt 25 kg, maar is redelijk compact (32 x 18 x 17 cm). Hierdoor is hij lastig te verplaatsen zonder je vingers te bezeren. Hij staat niet alleen voor de elektrische veiligheid op 4 rubberen voetjes, dat hoor je direct als je hem inschakelt. Geen dreun, maar een constante brom die mensen gek zou maken en niet alleen de kantoormedewerkers in de nabijheid. Van wat ik lees op het web is het gebrom het gevolg van het feit dat de transformatoren expres met hoge ijzerverliezen worden gebruikt om in resonantie te blijven op 50 Hz in combinatie met de condensatoren. Hoe dat een stabiliserend effect van 1% oplevert, moet ik nog ontdekken, maar het systeem werkt als een speer en onderdrukt alles dat geen 50 Hz is. Ik kwam daar achter door de PE1033 thuis aan te sluiten (waar ik meestal 230-235 VAC meet) en de vorm van het uitgangssignaal te bekijken. Vergeleken met wat erin ging, was de uitvoer onberispelijk schoon. Tijdens de meting werd de netspanning op mijn werktafel zwaar geteisterd door uitbarstingen en stroomstoten van mijn 7-kWp zonnecel-installatie met slechts 4,8 kWp op een gedeeltelijk bewolkte middag.
Ondenkbaar tegenwoordig is het vermogensverlies dat dit type netstabilisator wordt: bij een belasting van 105 watt (een Tektronix 310A scoop) heb ik ongeveer 280 watt aan ingangsvermogen gemeten. De stabiliteit en zuiverheid van de wisselspanning kost wat ... heel veel brom en warmte 24/7!
Mijn PE1033 heeft een Belgisch, mogelijk Frans stopcontact, dat geïmproviseerd vastgemaakt is aan de achterkant van de kast (**figuur 6**). Dit is aangesloten op de OUTPUT-aansluitingen van de PE1033 (L en N) en vastgeschroefd aan het chassis (GND). Zowel de INPUT- als de OUTPUT-aansluitpool bevinden zich in de behuizing en zijn bedoeld voor een permanente aansluiting (**figuur 7**).
Stabiliseren, filteren en brommen kan de PE1033 als de beste — maar nog steeds biedt hij niet de fel begeerde volledige elektrische isolatie van het lichtnet. Bovendien heeft hij bepaald geen uitgesproken instelbereik. Voor deze en andere desiderata moeten we verder kijken.

Instelbare transformator - geen opdonder meer van de tv

Op onze werktafel willen we ten minste één stopcontact met een wisselstroombron die zowel over een groot bereik instelbaar (bv. 10-260 V), als dubbel geïsoleerd van het lichtnet is, met een echte transformator met een primaire en een secundaire wikkeling dus, geschikt voor een zeer hoge isolatiespanning tussen de wikkelingen (pakweg 2 kV). Met een zekering... en stil. Een apparaat dat aan deze vereisten en meer voldoet, bestaat. Onlangs kreeg ik er een van een oude radio- en tv-reparateur die in de jaren zeventig zijn baas ervan overtuigde een Grundig RT 5A te installeren (**figuur 8**). Dit was het neusje van de zalm wat betreft elektrische veiligheid en veelzijdigheid in

het begin van de jaren zeventig. Maar het is met zijn 18 kg wederom een zwaar apparaat.
Het Duitse Grundig was een belangrijke fabrikant van radio- en tv-toestellen en de RT-serie (**R***egel*-**T***renn-Transformator*) werd vaak gezien in RTV-reparatiewinkels. De mijne is een 3,5A-type (zeg 800 VA) met een aparte uitlezing voor spanning en stroom met draaispoelmeters.
De symbolen op het voorpaneel van de RT 5A (**figuur 9**) zijn belangrijk en weerspiegelen de Duitse *Gründlichkeit*: dubbele isolatieklasse, gezekerde en volledig gescheiden transformatorwikkelingen, aan / uit-schakelaar.
De hoofdreden voor webdiscussies over de RT 5A is dat de kans heel groot is dat uw huiszekering er aan gaat als de RT 5A wordt ingeschakeld. Deze zekeringen zijn nu vaak semi-automatisch met een hoge of gemiddelde snelheid en hebben een snellere respons dan de 16A porselein/draadzekeringen die in de jaren zeventig werden gebruikt. Mijn RT 5A is geen uitzondering. Ook al heeft hij een langzame zekering van 10 A op het voorpaneel, een inschakelvertraging die een grote serieweerstand (bv. 10 ohm, 25 watt) in serie met de primaire transformator zet, is geen luxe. De weerstand beperkt de inschakelstroom de eerste 500 ms na het inschakelen en wordt dan overbrugd door een relais. De RT 5A continu aan laten staan is geen optie, want hij bromt en verspilt veel energie.
Als je de RT 5A openmaakt, is de montagemethode van de elektronica dik in orde, maar de mechanische constructie is veel te zwak voor een transformator van meer dan 15 kg. Om het gewicht te beperken, hebben de ontwerpers van Grundig op de dikte van het plaatstaal bezuinigd — het is veel te dun voor dat doel.

Conclusie

Elk instrument heeft zijn eigen voordelen en tekortkomingen, alles afhankelijk van wat je ermee wilt doen en waar.
De quasi-draagbare variac is geweldig voor 'mobiel gebruik'. Het vermogen van 800 watt volstaat voor de meeste onbekende apparaten die ik gecontroleerd wil opstarten. De vaste 115 VAC-uitgang is handig voor Amerikaanse apparatuur; helaas ontbreekt een stroommeter om problemen aan te geven voordat de eerste rook verschijnt.
De PE1033 is een dinosaurus die alleen uitblinkt in het filteren van de netspanning. De stabilisatie van de uitgangsspanning binnen 1% was destijds van het hoogste belang, maar dit is tegenwoordig van beperkt nut. Mogelijk is hij nog inzetbaar als benchmark of *known-good-boatanchor* (KGB).
De RT 5A is mijn favoriet, hij biedt volledige elektrische isolatie en is instelbaar van 0 tot 250 VAC. Gezien de brakke behuizing is hijniet zo geschikt om mee te nemen zoals de variac. Voor hedendaags gebruik vereist de RT 5A een inschakelvertraging! Ik heb ze toevallig alle drie in de Retrotronica-collectie, en dus tot mijn beschikking als de situatie daar om vraagt, verkregen voor de prijs van schroot. Geen opgeblazen zekeringen meer!

180574-02

Figuur 8. De Grundig Electronic RT 5A uit ongeveer 1970 combineert een variac met een instelbare transformator, wat betekent dat de uitgang ervan is geïsoleerd van het net. Dit is geweldig voor reparaties aan oude tv-toestellen en radio's waar het chassis aan de fase (L) kan hangen. Merk op dat de stroomaansluiting op het voorpaneel niet geaard is en dat is zowel verstandig als wenselijk.

Figuur 9. Wie heeft een gebruikershandleiding of een schakelschema nodig als alles wat u moet weten van de Grundig RT 5A-variac / isolator is afgedrukt met symbolen op het voorpaneel?

Figuur 10. Twee kleine stickers die tijdens het openen van de behuizing uit de RT 5A zijn gevallen, vatten de elektrische samenstelling van het instrument samen.

ESTD 2004

www.elektor.tv

Retrotronica is een rubriek over oude elektronica en legendarische Elektor-ontwerpen. Bijdragen, suggesties en vragen zijn meer dan welkom; stuur uw telex of telegram naar **redactie@elektor.nl**

Het Interplanetary File System

achter de schermen wordt gewerkt aan een nieuw wereldwijd web

Tessel Renzenbrink (Nederland)

Een Aardse verkenner kruipt op dit moment op Mars rond. NASA's Curiosity maakt deel uit van de robot-voorhoede die de rode planeet onderzoekt. Het doel is dat die robots op een dag gezelschap zullen krijgen van mensen. En zodra de eerste interplanetaire pioniers zich op Mars vestigen, zullen zij ook door Wikipedia willen kunnen bladeren.

Bron: NASA

Maar met de huidige structuur van het wereldwijde web krijgen ze dan onherroepelijk met het probleem van de latentietijd (*latency*) te maken. Een Martiaanse webbrowser doet er minimaal 4 minuten over om verbinding te maken met de Wikipedia-server op aarde. En afhankelijk van de stand van de planeten kan die latency oplopen tot 24 minuten.

Het Interplanetary File System (IPFS) wil dit oplossen door het web op een nieuw protocol te stoelen. IPFS lost daarmee niet alleen een hypothetisch toekomstig probleem op. Het latency-probleem geldt vandaag al voor mensen die op grote afstand van datacenters wonen of langzame verbindingen hebben zoals in landelijke gebieden en ontwikkelingslanden. IPFS is niet alleen sneller dan het huidige web. Het lost ook andere euvels op die het web op dit moment teisteren, zoals censuur, privacyschendingen en DDoS-aanvallen (*Distributed Denial of Service*). Dat klinkt haast te mooi om waar te zijn. Om te begrijpen waar deze claim op is gebaseerd, is het zinvol te kijken naar wat er mis is met het huidige web.

De kwalen van het wereldwijde web

Op dit moment is het web een sterk gecentraliseerd systeem (**figuur 1a**). In een gecentraliseerd computernetwerk is er één centraal knooppunt. Alle communicatie tussen computers in het netwerk loopt via dat centrale knooppunt. De zwakte van deze configuratie is direct zichtbaar. Het netwerk is kwetsbaar omdat het niet meer kan functioneren als het centrale knooppunt wegvalt.

Privacyschendingen

Naast de kwetsbaarheid van het systeem leidt centralisatie ook tot een ongelijke machtsverhouding. Het centrale knooppunt is heer en meester over de informatiestromen. Het kan weigeren om data door te geven, zelf data lezen of tol heffen op de doorvoer van data. Die macht kan leiden tot privacyschendingen. We hangen bijvoorbeeld *en masse* aan de servers van Facebook en Google. De techreuzen positioneren zichzelf tussen de content op het web en de mensen die die content willen benaderen. Denk bijvoorbeeld aan Youtube. Google maakt zelf geen video's; dat doen miljoenen mensen wereldwijd. Dat was ook de oorspronkelijke gedachte achter het wereldwijde web: mensen die content en informatie via het web aan anderen beschikbaar stellen. Google en Facebook hebben zich als centraal punt tussen de content en de mensen gemanoeuvreerd. De tol die bezoekers betalen bestaat uit hun persoonlijke gegevens.

Censuur

De centralisatie van het web is een gevolg van het client-server model waarop het gebaseerd is (**figuur 2**). Wie bijvoorbeeld Wikipedia wil bezoeken, moet contact maken met een server waarop de website wordt gehost. Als de toegang tot die server geblokkeerd wordt, is de informatie voor de cliënten niet meer beschikbaar. Dit gebeurde bijvoorbeeld in 2017 in Turkije, toen de overheid daar Wikipedia in de ban deed.

DDoS-aanvallen

Het client-server model is ook de reden waarom DDoS-aanvallen werken. Simpel gezegd is een DDoS-aanval een gigantisch aantal informatieverzoeken aan een server. Die server wordt dan overspoeld met aanvragen. Hierdoor kunnen mensen met legitieme informatieverzoeken de server niet meer bereiken. Alweer geldt dat het web kwetsbaar is omdat informatie slechts één centrale locatie opgehaald kan worden.

De belofte van het IPFS

Zoals gezegd komt IPFS met grote beloften. Het is sneller, censuurbestendig en geeft ons controle over onze data. Dit komt omdat IPFS geen gecentraliseerd maar een gedistribueerd netwerk is. Een gedistribueerd netwerk (**figuur 1c**) is verreweg de veerkrachtigste netwerkconfiguratie. Elke computer in het netwerk fungeert als knooppunt waarover informatie kan stromen. Als een aantal computers uitvalt, kan de rest van het netwerk gewoon blijven functioneren. In een gedistribueerd netwerk is er geen hiërarchie: alle knooppunten in het netwerk zijn gelijk.

Het gedistribueerde IPFS-netwerk is niet gebaseerd op het client-server model voor het uitwisselen van data. In plaats daarvan worden bestanden *peer-to-peer*

Figuur 1. Dit schema is bedacht door elektrotechnicus Paul Baran voor zijn onderzoek naar veerkrachtige communicatienetwerken. Het werd in 1962 gepubliceerd in de paper 'On Distributed Communications Networks' door de Rand Corporation (https://bit.ly/2AFAqdp).

gedeeld. Dat wil zeggen dat elke computer in het netwerk een bestand kan hosten. Wie dat bestand wil downloaden, vraagt om een kopie van de dichtstbijzijnde hosts. Vervolgens kan de downloader zelf ook een host worden en het bestand verder helpen verspreiden (**figuur 3**).

De onzichtbare overname van het web

Door deze en andere technische eigenschappen van het IPFS is het niet vatbaar voor de kwalen waar het huidige web mee kampt. DDoS werkt niet omdat er geen centrale server is om aan te vallen. Dat is ook de reden dat IPFS bestand is tegen censuur. Tijdens de Turkse Wikipedia-ban werd een kopie van Wikipedia beschikbaar gesteld via een IPFS-netwerk. Omdat de informatie daarbij op meerdere computers gehost wordt, is het veel moeilijker de verspreiding ervan te onderdrukken.

IPFS geeft mensen ook controle over hun data. Wie een video wil delen, kan deze op het netwerk beschikbaar stellen, bijvoorbeeld door het op de eigen computer te hosten, waarna die verder in het netwerk verspreid kan worden. De content is direct beschikbaar en is niet opgesloten in platformen als Youtube en Facebook. Tenslotte werkt IPFS ook offline. Het systeem hoeft niet constant met het complete fysieke internet in verbinding te staan om te functioneren. Het werkt ook lokaal. Dat is ook de reden waarom de toekomstige Marsbewoners beter af zijn met IPFS. De eerste Martiaan die Wikipedia wil bekijken, zal het van het Aardse netwerk moeten plukken en de lange wachttijd op de koop toe moeten nemen. Maar daarna hoeft geen van zijn kompanen de langzame route naar Aarde meer af te leggen. Wikipedia is nu lokaal op Mars beschikbaar.

Misschien wel het meest indrukwekkende aspect van IPFS is dat de bedenkers het willen doorvoeren zonder dat iemand het door heeft. Onzichtbaar voor de eindgebruikers wordt het huidige HTTP-protocol vervangen door IPFS. Juan Benet, die aan de wieg van IPFS stond, zei daarover in een presentatie voor studenten aan de Stanford University: 'Als niemand weet dat er een switch heeft plaatsgevonden, dan hebben we gewonnen. Het doel is om dezelfde interface te behouden. Als je namelijk een systeem ontwerpt dat gebruikers dwingt hun gewoontes te veranderen, dan zullen ze het domweg niet gebruiken. Dat is de manier waarop het internet verbeterd moet worden: niemand moet het doorhebben. *It should just get better*.' [1]

180572-01

IPFS is een open source-project dat iedereen kan gebruiken en waar iedereen aan kan bijdragen. Voor meer informatie: https://ipfs.io.

Figuur 2. Het client-server model (David Vignoni: https://bit.ly/2t7aPmM).

Figuur 3. Bittorrent-netwerk (Scott Martin: https://bit.ly/2PCnuNu).

Weblink

[1] Stanford Seminar – IPFS and the Permanent Web, door Juan Benet.: https://bit.ly/2qncrtx

welkom in je WEB STORE

TIP VAN DE REDACTIE

De hedendaagse elektronica-ontwikkelaar en elektronica-reparateur dromen nog wel eens van een thermobeeldcamera. Alleen liggen de meeste exemplaren in een prijscategorie die het serieus moeilijk maakt een dergelijke aanschaf te verantwoorden. Maar langzaam komt daar verandering in. Op ons eigen lab gebruiken we de HT-02 Thermal Imaging Camera. En daar zijn we heel tevreden over. De instelmogelijkheden zijn erg compleet. Toegeven, ik ben misschien geen specialist op het gebied van thermische camera's, maar voor de doelgroep — reparateurs en ontwikkelaars — kan ik in de gauwigheid geen ontbrekende instellingen verzinnen die onontbeerlijk zijn in deze prijsklasse. Hij doet eigenlijk prima wat hij moet doen .

Thijs Beckers (Redactie Elektor)

www.elektor.nl/ht-02

Elektor Bestsellers

1. Raspberry Pi 3 (model A+)
   www.elektor.nl/rpi3a-plus

2. Android App Development
   www.elektor.nl/b4a-boek
3. ESP32 programmeren voor beginners
   www.elektor.nl/esp32-programmeren
4. VFD Tube Clock
   www.elektor.nl/vfd-tube-clock
5. Raspberry Pi 3 (model B+)
   www.elektor.nl/rpi3b-plus
6. Nixie Bargraph Thermometer
   www.elektor.nl/nixie-thermometer

Raspberry Pi Ontdekken

In dit boek staan de sterke kanten van de Raspberry Pi centraal: de combinatie van programmeren en elektronica. Na een korte introductie worden maar liefst 45 spannende en interessante projecten in detail besproken en uitgelegd. Van een wisselknipperlicht tot lichtmeter en temperatuurregeling en van een motorsnelheidsregeling tot een webserver met CGI.

Ledenprijs: € 35,96

www.elektor.nl/rpi-ontdekken

Programming with STM32 Nucleo Boards

met gratis L476RG BOARD

Dit (Engelstalige) boek bevat projecten met de STM32 Nucleo ontwikkelboards van STMicroelectronics. De projecten variëren van eenvoudige knipperende LED's tot complexere meer projecten. De meeste zijn gebaseerd op het Nucleo-L476RG board. Dit board (t.w.v. € 15) krijgt u tijdelijk gratis bij aankoop van het boek!

Ledenprijs: € 31,45

www.elektor.nl/stm32-boek

Porsche Carrera Race Engine

De Carrera 547 motor was de eerste Porsche motor speciaal ontwikkeld voor gebruik in races. De Carrera Race Engine kit brengt de 547 verticale as motor naar uw bureau als een transparant functioneel model op een schaal van ongeveer 1:3. Met dit pakket, dat meer dan 300 onderdelen bevat, kunt u (zonder lijm) in ongeveer zes uur een transparant functioneel model van de viercilinder boxermotor uit 1953 bouwen.

Ledenprijs: € 206,10

www.elektor.nl/carrera

nieuw

Electronics for Space

De ruimtevaartindustrie groeit voortdurend en er zullen nieuwe producten en diensten nodig zijn. Voor de ontwikkeling van deze industrie is innovatie nodig. Vandaag de dag is het haast niet meer mogelijk om alle gebeurtenissen op het gebied van de ruimtevaart te volgen. De activiteiten nemen toe, met name op de markt voor kleine satellieten. Een van de grootste problemen is het vinden van mensen met kennis van het ontwerpen van ruimte-elektronica. Dit (Engelstalige) boek wil helpen om de kloof te dichten en elektronische ingenieurs aan te moedigen om het fascinerende gebied van de ruimtevaartelektronica te betreden.

LEDENPRIJS: € 22,46
www.elektor.nl/electronics-for-space

ESP32 programmeren voor beginners

In dit boek bekijken we de ESP32 en zijn voorganger, de ESP8266. Deze kleine, maar zeer krachtige Chinese chips bestaan uit een processor, geheugen, input- en outputaansluitingen en hebben draadloze netwerktoegang via WiFi. Met dit boek tonen we hoe de programmeeromgeving Arduino werkt en hoe die gebruikt kan worden met ESP-hardware. Aan de hand van voorbeelden en projecten verduidelijken we hoe de ESP-hardware kan worden toegepast.

Ledenprijs: € 20,25

www.elektor.nl/esp32-programmeren

Raspberry Pi TV HAT

Met dit board kunt u televisie ontvangen en bekijken op een Raspberry Pi, of je kunt de Pi als een server gebruiken om televisie via een netwerk naar andere apparaten te streamen. De TV-HAT sluit u aan op de 40-pins GPIO-header en met een geschikte antenne, zodat uw Raspberry Pi DVB-T2-tv-uitzendingen kan ontvangen.

Ledenprijs: € 22,45

www.elektor.nl/raspberry-pi-tv-hat

Android App Development

Apps ontwikkelen om elektronica met je smartphone aan te sturen wordt steeds populairder. In dit Engelstalige boek presenteert Dogan Ibrahim de programmeertaal Basic For Android (B4A) en laat hij zien hoe je deze in eenvoudige elektronicaprojecten kunt toepassen. Het boek beschrijft verder hoe een Android mobiel apparaat kan communiceren met populaire hardware platforms via een Wi-Fi verbinding of door gebruik te maken van SMS tekstberichten.

Ledenprijs: € 26,96

www.elektor.nl/android-app-development

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 getallen 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 21 januari 2019** naar: **hexadoku@elektor.nl**

De prijswinnaars

De juiste oplossing van de Hexadoku uit het november/december-nummer is: **386BF**.
De Elektor-waardebonnen van 50 Euro zijn gewonnen door Frederik Andriof (Duitsland), Freddy Ghys (België), Biette Francis (Frankrijk), Alexandr Papazyan (Rusland) en Henk Brand (Nederland).
Allemaal van harte gefeliciteerd!

3	8		A	C		E			1		0	D		B	4
6	7	F	1		D	B			3	2		E	5	0	C
		E	B				7	D				F	6		
	C					6			F					9	
	A			6		D			4		2			5	
2			E	4	3		9	1		C	6	A			F
	9				C	7	0	3	E	5				8	
1	6		4		2					D		9		C	B
	E			0			5	2			F			A	
A	F		0	D		3			6		E	5		7	9
	D	7	C	9							3	B	0	F	
5		8			E					0			C		1
			F	5	7					E	4	0			
		6			A					3			F		
D		A		E							1		9		5
	2			B	F					9	5			1	

0	E	3	7	F	9	4	8	1	2	6	A	5	D	C	B
F	2	1	A	D	0	C	3	7	5	8	B	E	6	9	4
4	6	5	8	1	2	E	B	C	9	3	D	F	0	7	A
9	B	C	D	A	5	6	7	4	E	F	0	1	2	8	3
E	3	8	6	B	F	D	C	5	1	A	4	9	7	0	2
A	C	9	B	4	6	0	E	3	7	D	2	8	5	1	F
D	F	0	5	3	7	1	2	8	6	B	9	A	4	E	C
7	1	2	4	5	8	9	A	E	F	0	C	B	3	6	D
B	4	E	9	C	D	F	5	A	0	1	3	6	8	2	7
1	5	D	2	6	3	8	9	F	4	C	7	0	A	B	E
8	0	6	F	7	E	A	4	9	B	2	5	3	C	D	1
C	7	A	3	0	B	2	1	D	8	E	6	4	9	F	5
5	9	F	C	8	1	7	D	6	A	4	E	2	B	3	0
2	A	B	1	9	C	3	F	0	D	5	8	7	E	4	6
3	8	4	0	E	A	B	6	2	C	7	1	D	F	5	9
6	D	7	E	2	4	5	0	B	3	9	F	C	1	A	8

Medewerkers van Elektor International Media en hun familieleden zijn van deelname uitgesloten.

Controle over alles, van waar dan ook met gemoedsrust

Slimme, verbonden en veilige ontwerpen bouwen

www.microchip.com/SmartConnectedSecure

microchip DIRECT
www.microchipdirect.com

MICROCHIP

De naam en het logo van de Microchip en het logo van de Microchip zijn gedeponeerde handelsmerken van Microchip Technology Incorporated in de VS en andere landen. Alle andere handelsmerken zijn het eigendom van hun geregistreerde eigenaars.
© 2018 Microchip Technology Inc. All rights reserved. DS00002758A. MEC2220Dut08/18

reichelt elektronik

Technology connects.

- ✓ Meer dan 100.000 producten
- ✓ Geweldige prijs-kwaliteitverhouding
- ✓ Hoge beschikbaarheid en betrouwbare levering
- ✓ Sterke merken en voortreffelijke kwaliteit

HET BELANGRIJKSTE GEREEDSCHAP VOOR ELEKTROMONTEURS

Weller®

Het Weller WE1010 Education Bundle

70 watt vermogen voor professionele eisen!

Het sterkste soldeerstation in zijn klasse!
Met dit voordeelpakket bent u perfect uitgerust voor alle taken. Naast het WE1010 soldeerstation bevat deze set een 170 mm Weller Xcelite zijsnijtang, een rol Weller WSW SCN N1 soldeertin alsmede een extra soldeerpunt ETB 2,4 mm.

- ESD-veilig soldeerstation, soldeerbout en hitte-bestendige siliconenkabel
- Temperatuurstabiliteit en -vergrendeling beschermt de soldeerpunt en de componenten
- Wachtwoordbeveiliging slaat alle instellingen op

Weller Educational Bundle:
+ Zijsnijtang
+ Soldeertin
+ Soldeerpunt

Bestelnr.: WELLER WE1010EB

NIEUW 202,55 (€ 167,40)

TRMS-digitale-multimeter

XLABs

Met geïntegreerde led-werklamp.

- Verlicht display met 6.000 tellingen
- Spanning tot 1000 V DC/750 V AC
- Stroom max. 10 A AC/DC
- Frequentie-, capaciteits-, weerstandsmeting
- Duty-cycle-meting: 5–95 %

DE ALLROUNDER VOOR ALLE MEETTAKEN

EN 61010-1 CAT III 1000 V | EN 61010-1 CAT IV 600 V | EN 61010-1

Bestelnr.:
VAL MS0205

BEST SELLER 40,63 (€ 33,58)

6-delige set schroevendraaiers

RND lab

Ergonomisch gevormd voor uitstekend grip- en schroefcomfort.

- Messen: hoogwaardig chroom-vanadium-molybdeen-staal
- Mesprofielen:
  2x kruiskop:
  PH0/PH00
  4x sleuf:
  1,6/2,0/2,4/
  3,0 mm

Bestelnr.:
RND 550-00197

NIEUW 15,24 (€ 12,60)

Led-loeplamp met lens van echt glas

Schaduwvrije verlichting en gedetailleerde vergroting.

- Lens: Ø 100 mm, 3 dioptrieën
- Ringvormige diffusor
- Licht: koud wit (6.400 K)

A+
(A++ - E)

Bestelnr.:
LL ZD-2012B-2H

in plaats van 25,38

20 % BESPAREN 20,30 (€ 16,77)

Dagprijzen · Prijzenstand: 4. 12. 2018

Betaalwijzen: Sofort. VISA MasterCard Maestro PayPal iDEAL amazon pay

De wettelijke herroepingsregelingen zijn van toepassing. Alle aangegeven prijzen in euro inclusief de wettelijke BTW, excl. verzendkosten voor de totale winkelwagen. Uitsluitend onze Algemene Voorwaarden (zie hiervoor https://rch.lt./AV-NL of op aanvraag) zijn van toepassing. Afbeeldingen kunnen afwijken. Drukfouten, vergissingen en prijswijzigingen voorbehouden.
reichelt elektronik GmbH & Co. KG, Elektronikring 1, D-26452 Sande, Tel.:+49 (0)4422 955-360

www.reichelt.nl

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