

Arduino in control

physical computing met het Arduino-platform

Marc Friedheim

Elektor International Media BV
postbus 11
6114 ZG Susteren
www.elektor.nl

Enkele afbeeldingen in dit boek opgenomen met toestemming van of zijn afkomstig uit datasheets van Arduino™, Fritzing™, Mapper Lithography BV, Texas Instruments (L293), International Rectifier (IRL540), Allegro microsystems (A3982), Microchip (TC74).

Met dank aan Frans Friedheim, Lenie Friedheim, Els van Tol, Alexander Eigenraam, Tom Duivenvoorde, Tim Spruijt, Peter-Paul Crans, Irene van Roon, Rinze Mozes, klas 4m anno 2012 en klas 4m anno 2013, Arno van Amersfoort, Marien Spek.

Arduino in control – physical computing met het Arduino-platform
2^e druk: juli 2015

© 2014 Elektor International Media BV
website: www.elektor.nl

ISBN: 978-90-5381-280-8
NUR 950

Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand, of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door middel van druk, fotokopie, microfilm of op welke andere wijze dan ook, zonder voorafgaande schriftelijke toestemming van de uitgever. Alle rechten voorbehouden.

Ondanks alle aan de samenstelling van dit boek bestede zorg kunnen noch de auteur, noch de redactie, noch de uitgever aansprakelijkheid aanvaarden voor eventuele schade die zou kunnen voortvloeien uit enige fout die in deze uitgave zou kunnen voorkomen.

Voor zover het maken van reprografische verveelvoudiging uit deze uitgave is toegestaan op grond van artikel 16h Auteurswet 1912 dient men de daarvoor wettelijk verschuldigde vergoedingen te voldoen aan de Stichting Reprorecht (Postbus 3051, 2130 KB Hoofddorp, www.reprorecht.nl).

Voor het overnemen van gedeelte(n) uit deze uitgave in bloemlezingen, readers en andere compilatiewerken (artikel 16 Auteurswet 1912) kan men zich wenden tot de Stichting PRO (Stichting Publicatie- en Reproductierechten Organisatie, Postbus 3060, 2130 KB Hoofddorp, www.stichting-pro.nl).

No part of this book may be reproduced, stored in a database or retrieval system, or published in any form or any way, electronically, mechanically, by print, photoprint, microfilm or any other means without prior permission from the Publisher.

All rights reserved.

Bewerking, tekstredactie en illustraties: E.A.J. Bogers, Saarbrücken (Duitsland)
Omslagontwerp: Elektor International Media B.V.
Grafische vormgeving & productie: Technisch Vertaalbureau Bogers, Saarbrücken (Duitsland)
Druk: Wilco, Amersfoort

149015-1/NL

Inhoud

1	Introductie	11
1.1	Beste student	11
1.2	Beste docent	11
1.2.1	Didactisch model / leermethode	11
1.2.2	Metacognitie	12
1.2.3	Doorzettingsvermogen	13
2	Basiskennis	15
2.1	Controle door geavanceerde technologie	16
2.1.1	Uitvoeringsvormen van de microcontroller	17
2.2	Wat is physical computing?	18
2.3	De ATmega328-microcontroller	18
2.3.1	I/O, ADC, PWM en seriële bussen	19
2.3.2	CPU (Central Processing Unit)	21
2.3.3	Timers en counters	21
2.3.4	Geheugens van de Arduino-microcontroller ATmega328	21
2.4	Een project beginnen	22
2.4.1	Denken in blokschema's	23
2.4.2	Eén taak of meerdere taken?	24
2.4.3	Toepassingen	25
2.4.4	Voedingsspanning	25
3	Het Arduino UNO-platform	27
3.1	De IDE	27
3.2	De hardware	28
3.2.1	Andere boards	30
3.2.2	Shields	30
3.3	De programmeeromgeving	31
3.3.1	Setup	32
3.3.2	Loop	32
3.3.3	Constanten en variabelen	32
3.3.4	Bibliotheken	33
3.4	Stijl	34

4	Physical Computing – oefenen met uitleg	39
4.1	Onboard knipper-LED	41
4.2	Knipper-LED met externe LED	44
4.3	Knipper-LED met NOT-instructie	46
4.4	Knipper-LED met XOR-instructie	47
4.5	4-LED looplicht (3 manieren)	48
4.5.1	Alternatief 1	50
4.5.2	Alternatief 2	51
4.5.3	De for-constructie	52
4.6	Aan/uit-LED met drukknop	53
4.6.1	De if-constructie	55
4.6.2	De while-constructie	56
4.6.3	Contactdender	57
4.6.4	Zwevende ingangen	58
4.6.5	Contactdender voorkomen (alternatieve methode)	60
4.7	Relais	61
4.8	Serieel data verzenden naar de PC	64
4.9	De uitkomst van een optelling serieel verzenden naar de PC	67
4.10	Met AND en OR rekenen en serieel verzenden	68
4.11	Analoog inlezen en serieel verzenden naar de PC	70
4.12	Digitale temperatuursensor TC74	72
4.13	Analoog inlezen, PWM uitvoeren	76
4.13.1	Het analogWrite()-commando	77
4.14	Digitaal inlezen, PWM uitvoeren	79
4.15	Analoog inlezen, digitaal schakelen	81
4.16	Het EEPROM-geheugen	83
4.16.1	Toevalsgetallen en data loggen	85
4.17	Externe interrupt	86
4.17.1	De instructie attachInterrupt();	88
4.18	Servo	89
4.18.1	De map()-functie	92
4.19	Het LC-display	93
4.20	DC-motorsturing	97
4.21	IR-naderingssensor	101
5	Physical Computing – praktijkvoorbeelden met uitleg	103
5.1	Stappenmotor en display	103
5.2	De stappenmotordriver A3982	105
5.3	DC-motorsturing met positiebepaling	108
5.3.1	De kwadratuur-encoder	109
5.3.2	Encoderresolutie	112
5.4	DC-motor besturen met instelbare draairichting en snelheid	114
5.5	De schaatsronde-meetmachine	120

6	Theoretische verdieping: software	125
6.1	Talstelsels	125
6.1.1	Decimale getallen omrekenen naar binaire getallen	126
6.1.2	Binaire getallen omrekenen naar hexadecimale getallen	126
6.1.3	Decimale getallen omrekenen naar hexadecimale getallen	127
6.1.4	Hexadecimale getallen omrekenen naar decimale getallen	127
6.1.5	Hexadecimale getallen omrekenen naar binaire getallen	127
6.2	Datatypes	127
6.2.1	Datatype <i>boolean</i>	128
6.2.2	Datatype <i>byte</i>	129
6.2.3	Datatype <i>array</i> (int / string / char)	131
6.2.4	Datatype <i>signed integer</i> (of <i>word</i>)	133
6.2.5	Datatype <i>unsigned integer</i> (of <i>word</i>)	133
6.2.6	Datatype <i>signed long</i>	134
6.2.7	Datatype <i>unsigned long</i>	135
6.2.8	Datatype <i>float</i>	135
6.2.9	Datatype <i>double</i>	137
6.2.10	Casting	138
6.3	Kiezen op basis van booleaanse operatoren	139
6.4	Bitmanipulaties (1)	140
6.5	Bitmanipulaties (2)	141
6.5.1	AND (&)	142
6.5.2	OR ()	143
6.5.3	XOR (^)	143
6.5.4	NOT (!)	144
6.5.5	shift left (<<) en shift right (>>)	144
6.6	Bewerkingen met getallen	145
6.6.1	Toevalsgetallen	145
6.7	Vergelijking van getallen	145
6.8	Samengestelde opdrachten	146
6.9	Wiskundige opdrachten en bewerkingen	146
6.10	Functies	150
6.10.1	Voorbeeld 1: functie zonder geretourneerde waarde en zonder parameter	151
6.10.2	Voorbeeld 2: functie zonder geretourneerde waarde maar met parameter	152
6.10.3	Voorbeeld 3: functie met geretourneerde waarde maar zonder parameter	153
6.10.4	Voorbeeld 4: functie met geretourneerde waarde en met parameters	154
6.11	Data control-structuren	155
6.11.1	if	155
6.11.2	if...else / else if	156
6.11.3	for	156
6.11.4	while	157

6.11.5	do...while	157
6.11.6	break	158
6.11.7	switch / case	158
6.11.8	millis	159
6.11.9	delay	160
6.11.10	delayMicroseconds	160
6.12	Constanten en variabelen	160
6.12.1	Constanten	161
6.12.2	Variabelen	161
6.12.3	Globale variabelen	162
6.12.4	Lokale variabelen	162
7	Problemen oplossen	165
7.1	Debuggen software	166
7.1.1	Foutzoeken software	168
7.1.2	Fouten bij datatypen	169
7.1.3	Fouten bij data control-structuren	169
7.2	Debuggen hardware	169
7.3	Fouten bij het aansluiten van hardware	170
8	Theoretische verdieping: hardware	173
8.1	I/O	173
8.2	Instelling van een I/O-pen	174
8.3	Digitale ingang	176
8.4	Analoge ingang	177
8.4.1	Referentiespanning	178
8.5	Digitale uitgang	179
8.6	PWM-uitgang	180
8.7	Interrupts	183
8.7.1	Hardwarematige interrupts	183
8.7.2	Softwarematige interrupts	183
8.8	Seriële bussen	186
8.8.1	Seriële communicatie en RS232	187
8.8.2	Baudrate	190
8.8.3	TWI / I ² C	191
8.8.4	SPI	191
8.9	Meer over het binnenste van de microcontroller	192
8.9.1	Architectuur, instructieset en operationele code	192
8.9.2	De taal 'C'	193
8.10	Bootloader	194
8.11	Verwerkingsnelheid	194
8.12	USB-communicatie	195
8.13	Geheugen	196
8.13.1	Flash-geheugen	197
8.13.2	SRAM-geheugen	197

8.13.3	EEPROM-geheugen	198
8.14	Program Counter	198
8.15	Functies en de stack	199
8.16	Structuur van de ATmega328	200
8.17	Het productieproces van een microcontroller	201
8.17.1	Gemaakt uit zand...	201
8.17.2	P-type en N-type	202
8.17.3	De productie van chips	202
9	Bijlagen	207
9.1	Het breadboard	207
9.2	LED's	208
9.3	Pull up- en pull down-weerstanden	209
9.4	De spanningsdeler	211
9.5	Motorsturing met de L293	212
9.5.1	DC-motorsturing	213
9.5.2	Stappenmotorsturing	214
9.6	Power-MOSFET	214
9.7	De ASCII-tabel	217
9.8	Lijst van afkortingen	218
9.9	Lijst van figuren	219
9.10	Lijst van tabellen	221
9.11	Lijst van codes	222
	Index	223

1

Introductie

1.1 Beste student

Je bent begonnen in een boek waarin uitgelegd wordt hoe je je eigen aansturingen kunt maken! Je leert in snel tempo niet alleen hoe je een lampje kunt laten knipperen, maar ook hoe je displays en verschillende typen motoren kunt besturen. Je staat op het punt een nieuwe wereld aan mogelijkheden te ontdekken met een heel modern microcontrollerplatform. De kracht hiervan zit in de software. Bibliotheekfuncties helpen de gebruiker door een stukje code uit te voeren zonder dat men die zelf hiet te programmeren. Er gebeuren een heleboel dingen zonder dat je dat allemaal zelf hoeft te programmeren. Er gebeuren dus dingen op de achtergrond, zonder dat je hier een heleboel code voor hoeft te typen. Dit betekent dat je heel snel stappen kunt maken. Dat is natuurlijk leuk. Omdat in dit boek ook wordt uitgelegd 'waarom' het zo werkt, leer je de achtergrond te begrijpen. Ik wens je heel veel succes met het 'leren en experimenteren'.

1.2 Beste docent

U wilt uw studenten theoretische en praktische kennis laten opdoen over microcontrollers en physical computing. Dan is dit boek een uitstekende keuze. Hieronder wordt het ween en ander verteld over de opzet en de insteek van het boek. Ik wens u veel succes en vooral plezier met lesgeven.

Praktische lessen kunnen het beste worden uitgevoerd met kleine groepen leerlingen.

1.2.1 Didactisch model / leermethode

Dit boek is geschreven om doelgericht te leren werken met de ATMEGA328 microcontroller die gebruikt wordt op het Arduino UNO-platform. Het didactisch model komt van het afstudeeronderzoek van de auteur, waarbij is onderzocht hoe multidisciplinaire en abstracte lesstof het beste gedoceerd kan worden aan praktisch ingestelde studenten. De lesmethode is praktisch ingericht en daarnaast onderbouwd met theoriestukken waarnaar wordt verwezen.

Door de vereiste abstractere kennis uit te leggen en toe te passen aan de hand van concrete, actuele praktijkvoorbeelden, ontstaat er een optimale mix van theorie en praktisch toepasbare kennis. Dit wordt ook wel het context–concept-lesmodel genoemd. Daarbij kan concreet worden samengewerkt met anderen in projectvorm. De nadruk van deze manier van werken ligt op het eindresultaat en succes waarbij doorzettingsvermogen een belangrijke rol speelt.

Deze specifieke leermethode verklaart waarom de opbouw van dit boek afwijkt van traditionele literatuur, die in eerste instantie gericht is op de theoretische onderbouwing. De opbouw die in dit boek gehanteerd wordt is:

- 1 Een introductie over microcontrollers in grote lijnen.
- 2 Basiskennis die nodig is om direct met het Arduino-platform aan de slag te kunnen.
- 3 Het vergaren van kennis en kunde aan de hand van op zichzelf staande oefeningen met uitleg.
- 4 Het vergaren van kennis en kunde aan de hand van geïntegreerde praktijkvoorbeelden met uitleg.
- 5 Theoretische verdieping.
- 6 Praktische elektronische schema's.

Het realiseren van een toepassing met geavanceerde componenten vereist niet alleen de nodige kennis over de interne werking hiervan, maar ook kennis en vaardigheid op het gebied van randapparatuur. U zult inzicht moeten krijgen in wat u wilt realiseren. Leer uzelf vanuit de voorbeelden in dit boek de beste methode aan om fouten op te sporen, zodat u hier tijdens grotere projecten op terug kunt vallen.

U zult merken dat er steeds een vertaalslag van uw eigen gedachten nodig is als u aan de gang gaat met het toepassen van deze cursus. Zaken die eerst onverklaarbaar lijken, blijken toch verklaarbaar.

1.2.2 Metacognitie

U zult er misschien tijdens het programmeren achter komen dat uw denkstappen niet altijd optimaal om te zetten zijn naar microcontroller-toepassingen. Veel factoren spelen een rol bij het realiseren van een toepassing. Bij deze toepassingen zijn er meerdere variabelen en onbekenden die in veel gevallen een snelle realisatie bemoeilijken. Daarnaast werkt het menselijk brein ook nog eens heel anders dan een microcontroller. Hierdoor zult u bekend moeten raken op onbekend terrein. Uw kennis over de eigen kennis (*metacognitie*) zal u steeds van pas komen tijdens dit leerproces en zal zich steeds uitbreiden na voltooiing en doorgronden van nieuwe projecten.

Tijdens het werken met microcontrollers en toepassingen zult u er ongetwijfeld achter komen dat het dikwijls niet allemaal direct zo gaat als u gedacht had. Er is één schrale troost: u kunt nooit de microcontroller de schuld geven. De microcontroller maakt namelijk geen fouten – zoals u waarschijnlijk wel zult gaan doen. Dat klinkt misschien

vervelend, maar het zou vervelender zijn als het omgekeerd was. Stel dat de microcontroller onbetrouwbaar zou zijn, dan is niet te herleiden waar de fout ligt. In dit boek staan voorbeelden die u stap voor stap wegwijs maken op dit gebied.

1.2.3 Doorzettingsvermogen

Er is doorzettingsvermogen nodig om uw toepassing te realiseren. Zeker in het begin kan er veel fout gaan. Elektronica kan verkeerd worden aangesloten, software kan verkeerd worden geprogrammeerd of u hebt de USB-kabel vergeten aan te sluiten. Van fouten leer je, mits je de fout vindt. Probeer daarom elke fout te herleiden, ook al kost dit veel tijd; de kennis die u opdoet vanuit de mislukkingen kunt u de volgende keer toepassen. Dit kan alleen als u weet *waar* het fout is gegaan. Verderop in het boek worden in het hoofdstuk 'Problemen oplossen' (bladzijde 165) hints gegeven hoe fouten op te sporen zijn.

Het Arduino-platform is vanwege zijn eenvoud, vergeleken met andere microcontrollerplatforms, uitstekend geschikt om mee te beginnen. Gelukkig betekent eenvoud in de bediening niet eenvoud in de mogelijkheden, en dat zult u hopelijk tijdens het oefenen in de volgende hoofdstukken ontdekken. Programma's kunt u, naarmate uw kennis en kunde zich ontwikkelen, krachtiger en efficiënter maken. Ook voor gevorderden blijft het een uitdaging om steeds nauwkeuriger, efficiënter en sneller toepassingen te realiseren. Daarbij spelen de manier van aanpak en denkwijze een belangrijke rol.