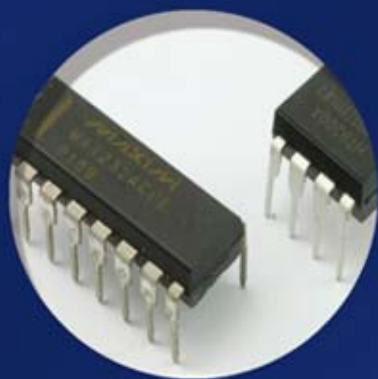
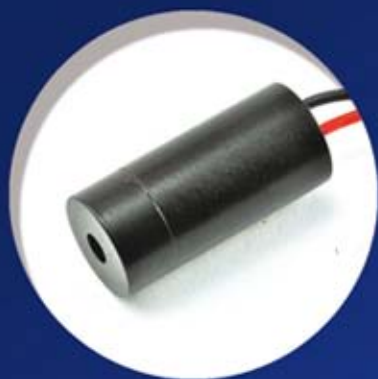




ZABAWY Z ELEKTRONIKĄ

Ilustrowany przewodnik
dla wynalazców i pasjonatów



Tytuł oryginału: Hacking Electronics: An Illustrated DIY Guide for Makers and Hobbyists

Tłumaczenie: Krzysztof Sawka

ISBN: 978-83-246-8051-1

Original edition copyright © 2013 by The McGraw-Hill Companies.
All rights reserved.

Polish edition copyright © 2014 by HELION S.A.
All rights reserved.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
<http://helion.pl/user/opinie/zabele>
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

O autorze	15
Wprowadzenie	17
ROZDZIAŁ 1. Pierwsze kroki	21
Czas na zakupy	21
Zakup elementów elektronicznych	22
Szukanie dawców	23
Podstawowe narzędzia	23
Usuwanie izolacji z przewodu	25
Co będzie potrzebne	26
Łączenie przewodów poprzez skręcanie	28
Co będzie potrzebne	28
Łączenie przewodów poprzez lutowanie	29
Bezpieczeństwo	29
Co będzie potrzebne	30
Lutowanie	31
Łączenie przewodów	32
Testowanie połączenia	33
Co będzie potrzebne	33
Wiatrak komputerowy jako wentylator odprowadzający toksyczne gazy	35
Co będzie potrzebne	35
Konstrukcja	36
Etap 1.: Usuwanie izolacji z przewodów zasilacza	37
Etap 2.: Identyfikacja biegunowości przewodów zasilających	37
Etap 3.: Łączenie przewodów masowych	38

6 Zabawy z elektroniką

Etap 4.: Łączenie przewodu prądowego z przełącznikiem	39
Etap 5.: Testowanie	40
Podsumowanie	40
ROZDZIAŁ 2. Teoria i praktyka	41
Zebranie początkowego zestawu elementów	41
Co będzie potrzebne	42
Identyfikowanie elementów elektronicznych	43
Rezystory (oporniki)	43
Kondensatory	45
Diody	47
Diody LED	47
Tranzystory	49
Układy scalone	49
Inne elementy	50
Elementy do montażu powierzchniowego	50
Natężenie, rezystancja, napięcie	51
Natężenie	51
Rezystancja	51
Napięcie	52
Prawo Ohma	52
Moc	54
Czytanie schematów	55
Pierwsza zasada schematów: Dodatnie napięcie znajduje się na górze	56
Druga zasada schematów: Wszystko dzieje się w prawą stronę	56
Nazwy i wartości	56
Symbole podzespołów	57
Podsumowanie	58
ROZDZIAŁ 3. Podstawowe sztuczki	59
Rozgrzewanie rezystora	59
Co będzie potrzebne	59
Eksperyment	60
Rezystory jako dzielniki napięcia	61
Co będzie potrzebne	61
Przekształcanie rezystancji na napięcie (i tworzenie światłomierza)	64
Co będzie potrzebne	64
Przekształcanie wciskanej lampki w czujnik oświetlenia	66
Co będzie potrzebne	67
Płytką montażowa	68
Konstrukcja	70

Wybór tranzystora bipolarnego	75
Karty katalogowe	75
Tranzystory MOSFET	76
Tranzystory PNP oraz z kanałem typu N	77
Popularne modele tranzystorów	78
Tranzystor mocy MOSFET jako regulator pracy silnika	78
Co będzie potrzebne	79
Płytki montażowa	79
Wybór właściwego przełącznika	81
Przełączniki wciskane	82
Mikroprzełączniki	83
Przełączniki dwustabilne	83
Podsumowanie	85
ROZDZIAŁ 4. Diody LED	87
Ochrona diody LED przed przepaleniem	87
Co będzie potrzebne	88
Diody	88
Diody LED	89
Testowanie	91
Wybór właściwej diody LED do pracy	92
Co będzie potrzebne	92
Jasność i kąt świecenia	93
Wielokolorowość	93
Diody świecące w podczerwieni i ultrafiolecie	94
Diody LED przeznaczone do oświetlania	95
Układ LM317 jako sterownik prądu stałego	96
Co będzie potrzebne	96
Projekt	97
Płytki montażowa	98
Konstrukcja	99
Pomiar napięcia przewodzenia diody LED	100
Co będzie potrzebne	102
Zasilanie większej liczby diod LED	102
Błyskające diody LED	103
Co będzie potrzebne	104
Płytki montażowa	104
Zastosowanie płytki perforowanej na przykładzie nadajnika błysków	106
Rozmieszczanie podzespołów na płytce perforowanej	107
Co będzie potrzebne	109
Konstrukcja	109
Rozwiązywanie problemów	113

8 Zabawy z elektroniką

Korzystanie z modułu diody laserowej	114
Modyfikacja zdalnie sterowanego samochodu wyścigowego	115
Co będzie potrzebne	115
Przechowywanie ładunku w kondensatorze	116
Projekt	117
Konstrukcja	118
Testowanie	119
Podsumowanie	119
ROZDZIAŁ 5. Baterie i zasilanie	121
Wybór właściwej baterii	121
Pojemność baterii	121
Maksymalna wartość rozładowania	122
Baterie jednorazowe	122
Akumulatorki	125
Ładowanie baterii (ogólne zasady)	127
C	128
Przeciążenie (zbyt mocne naładowanie)	128
Przeciążenie (zbyt pełne rozładowanie)	128
Żywotność baterii	128
Ładowanie ogniwa NiMH	129
Proste ładowanie	129
Szybkie ładowanie	130
Ładowanie zamkniętego akumulatora kwasowo-ołowiowego	130
Ładowanie za pomocą programowalnego źródła prądu	131
Ładowanie akumulatora Li-Po	132
Modyfikowanie baterii do telefonu komórkowego	133
Regulacja napięcia na baterii	134
Co będzie potrzebne	136
Płytki montażowa	136
Podwyższanie napięcia	137
Obliczanie czasu działania baterii	138
Projektowanie zasilania awaryjnego na baterie	139
Diody	139
Ładowanie podtrzymujące	141
Stosowanie ogniw słonecznych	142
Testowanie panelu słonecznego	143
Ładowanie podtrzymujące za pomocą panelu słonecznego	144
Zmniejszanie zużycia prądu	145
Podsumowanie	146

ROZDZIAŁ 6. Zabawy z Arduino	147
Konfiguracja układu Arduino (oraz migotanie diody LED)	148
Co będzie potrzebne	149
Konfiguracja układu Arduino	149
Modyfikowanie szkicu Blink	152
Arduino jako kontroler przekaźnika	154
Przekaźniki	155
Wyjścia układu Arduino	155
Co będzie potrzebne	156
Konstrukcja	157
Oprogramowanie	158
Obsługa zabawki za pomocą układu Arduino	159
Co będzie potrzebne	160
Konstrukcja	160
Monitor portu szeregowego	160
Oprogramowanie	162
Pomiar napięcia za pomocą Arduino	163
Co będzie potrzebne	164
Konstrukcja	164
Oprogramowanie	165
Arduino jako regulator diody LED	166
Co będzie potrzebne	167
Konstrukcja	167
Oprogramowanie (migotanie)	168
Oprogramowanie (jasność)	169
Odtwarzanie dźwięków za pomocą Arduino	170
Co będzie potrzebne	171
Konstrukcja	171
Oprogramowanie	171
Zastosowania modułów rozszerzeń układu Arduino	173
Sterowanie przekaźnikiem z poziomu strony internetowej	174
Co będzie potrzebne	175
Konstrukcja	176
Konfiguracja sieci	177
Testowanie	177
Oprogramowanie	178
Obsługa modułu alfanumerycznego wyświetlacza LCD	182
Co będzie potrzebne	183
Konstrukcja	183
Oprogramowanie	183

10 Zabawy z elektroniką

Napędzanie serwomotoru za pomocą Arduino	185
Co będzie potrzebne	185
Konstrukcja	186
Oprogramowanie	186
Charliepleksing diod LED	187
Co będzie potrzebne	189
Konstrukcja	189
Oprogramowanie	190
Automatyczne wpisywanie haseł	192
Co będzie potrzebne	192
Konstrukcja	193
Oprogramowanie	193
Podsumowanie	194
ROZDZIAŁ 7. Zabawa z modułami	195
Moduł wykrywacza ruchu na podczerwień	196
Co będzie potrzebne (czujnik ruchu i dioda LED)	196
Płytki montażowa	196
Co będzie potrzebne (czujnik PIR i Arduino)	198
Konstrukcja	198
Oprogramowanie	199
Moduły dalmierzy ultradźwiękowych	200
Co będzie potrzebne	202
Dalmierz HC-SR04	202
Dalmierz MaxBotix LV-EZ1	205
Moduł bezprzewodowego pilota	207
Co będzie potrzebne	208
Płytki montażowa	208
Moduł bezprzewodowego pilota i Arduino	209
Co będzie potrzebne	209
Oprogramowanie	210
Regulacja obrotów silnika za pomocą tranzystora mocy MOSFET	212
Co będzie potrzebne	212
Płytki montażowa	213
Oprogramowanie	213
Regulacja silnika stałoprądowego za pomocą modułu mostka H	215
Co będzie potrzebne	219
Płytki montażowa	219
Korzystanie z nóżek sterujących	220

Sterowanie silnikiem krokowym za pomocą modułu mostka H	221
Co będzie potrzebne	223
Konstrukcja	223
Oprogramowanie	223
Prosty łazik elektroniczny	226
Co będzie potrzebne	227
Konstrukcja	227
Testowanie	230
Oprogramowanie	230
Moduł siedmiosegmentowego wyświetlacza LED	232
Co będzie potrzebne	234
Konstrukcja	234
Oprogramowanie	234
Moduł zegara czasu rzeczywistego	236
Co będzie potrzebne	237
Konstrukcja	238
Oprogramowanie	239
Podsumowanie	240
ROZDZIAŁ 8. Zabawa z czujnikami	241
Wykrywanie toksycznego gazu	241
Co będzie potrzebne	242
Komparator LM311	242
Płytki montażowa	243
Wykrywacz gazu i Arduino	245
Pomiar barwy przedmiotu	246
Co będzie potrzebne	247
Konstrukcja	247
Oprogramowanie	248
Detekcja drgań	251
Co będzie potrzebne	251
Konstrukcja	252
Oprogramowanie	252
Pomiar temperatury	254
Co będzie potrzebne	254
Konstrukcja	254
Oprogramowanie	254
Akcelerometr	256
Co będzie potrzebne	257
Konstrukcja	258
Oprogramowanie	259

12 Zabawy z elektroniką

Wykrywanie pól magnetycznych	261
Co będzie potrzebne	261
Konstrukcja	261
Oprogramowanie	261
Podsumowanie	262
ROZDZIAŁ 9. Sztuczki z dźwiękiem	263
Modyfikowanie przewodów dźwiękowych	263
Ogólne zasady	264
Lutowanie przewodów dźwiękowych	265
Konwersja sygnału stereofonicznego na monofoniczny	266
Moduł mikrofonowy	268
Pluskwa radiowa	271
Co będzie potrzebne	271
Konstrukcja	271
Testowanie	273
Wybór głośników	274
Budowa jednowatowego wzmacniacza dźwięku	275
Co będzie potrzebne	277
Konstrukcja	277
Testowanie	278
Generowanie dźwięków za pomocą układu czasowego 555	279
Co będzie potrzebne	281
Konstrukcja	281
Kontroler muzyczny podłączany pod gniazdo USB	281
Co będzie potrzebne	281
Konstrukcja	282
Oprogramowanie	283
Programowy miernik głośności (VU)	284
Co będzie potrzebne	284
Konstrukcja	284
Oprogramowanie	285
Podsumowanie	286
ROZDZIAŁ 10. Psucie i naprawa	287
Jak uniknąć porażenia prądem?	287
Rozmontowywanie urządzeń ORAZ składanie ich z powrotem	289
Sprawdzanie bezpiecznika	289
Sprawdzanie baterii	291
Testowanie elementu grzejjego	291

Wykrywanie i wymiana uszkodzonych elementów	292
Sprawdzanie podzespołów	292
Wylutowywanie	293
Wymiana	294
Odzyskiwanie przydatnych elementów	294
Ponowne wykorzystanie ładowarki telefonicznej	295
Podsumowanie	296
ROZDZIAŁ 11. Narzędzia	297
Stosowanie miernika (ogólne)	297
Test ciągłości i diody	298
Rezystancja	298
Pojemność	298
Temperatura	299
Napięcie prądu zmiennego	300
Napięcie prądu stałego	301
Natężenie prądu stałego	301
Natężenie prądu zmiennego	302
Częstotliwość	302
Sprawdzanie tranzystora za pomocą miernika	302
Zasilacz laboratoryjny	303
Wprowadzenie do oscyloskopów	304
Oprogramowanie elektroniczne	305
Symulowanie	306
Fritzing	306
EAGLE PCB	307
Kalkulatory sieciowe	307
Podsumowanie	308
DODATEK Części	311
Narzędzia	312
Elementy elektroniczne	312
Startowe zestawy podzespołów	313
Rezystory	313
Kondensatory	314
Półprzewodniki	314
Sprzęt i pozostałe elementy	316
Moduły	318
Wykorzystywane w książce elementy zestawu Z1	
nieuwzględnione nigdzie indziej	320
Rezystory zawarte w zestawie Z2	320
Skorowidz	321

14 Zabawy z elektroniką

8

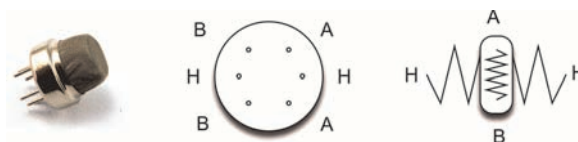
Zabawa z czujnikami

Rozdziały 6., 7. i 8. są ze sobą w pewien sposób powiązane, wiele czujników występuje w postaci modułowej, a moduły nieraz mogą być podłączane do Arduino.

W tym rozdziale nauczysz się korzystać z różnego rodzaju czujników sterowanych za pomocą układów elektronicznych, mikrokontrolera Arduino, a czasami obydwu mechanizmów naraz.

Wykrywanie toksycznego gazu

W tym podrozdziale będziemy się posługiwać wykrywaczem metanu (rysunek 8.1).



Rysunek 8.1. Wykrywacz metanu

Taki czujnik wydaje się drogi, ale w rzeczywistości jest naprawdę tani. Składa się z niewielkiego grzejnika (podłączonego pomiędzy dwoma mostkami H) oraz elementu katalitycznego, którego oporność zależy od stężenia metanu. Chociaż nasz układ będzie zasilany bateryjnie, trzeba będzie często wymieniać ogniwa, ponieważ element grzewczy czujnika pobiera prąd rzędu 150 – 200 mA.

Wykrywanie metanu okazuje się potrzebne w wielu dziedzinach nauki i przemysłu. My jednak wykorzystamy tę nowinkę techniczną w niecnym i infantylnym celu — do wykrywania pierdnieć.

Co będzie potrzebne

Do stworzenia wykrywacza gazu wykorzystamy następujące elementy:

Liczba	Oznaczenie	Przedmiot	Kod w dodatku
1	D1	Dioda LED	Z1
1	R1	Potencjometr 10 k Ω	Z1
1	R2	Rezystor 10 k Ω	Z2
1	R3	Rezystor 470 Ω	Z2
1	IC1	Komparator LM311	P7
1		Wykrywacz metanu MQ4	M11
1		Brzęczyk piezoelektryczny (z oscylatorem)	M10
1		Nielutowana płytki montażowa	N5
		Jednożyłowy przewód złączowy	N6
1		Uchwyt na 4 baterie AA	S1
1		4 baterie AA	
1		Zacisk na baterie	S2
1		* Arduino Uno/Leonardo	M2/M21
1		* Przewód USB: typu B dla Arduino Uno, microUSB dla Arduino Leonardo	

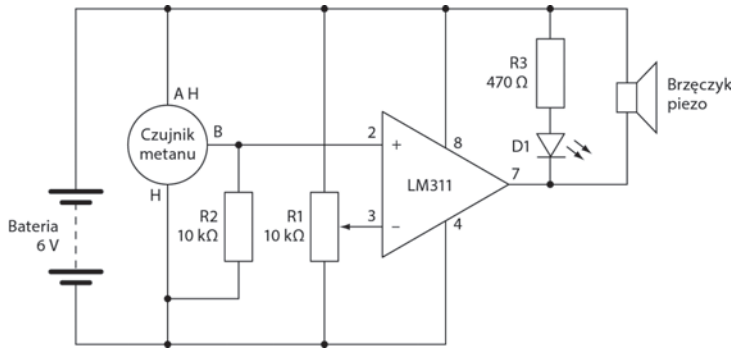
* Wymagane jedynie w przypadku, gdybyśmy chcieli podłączyć wykrywacz do układu Arduino.

Brzęczyk piezoelektryczny musi mieć własny obwód oscylatora oraz być zasilany napięciem 6 V.

Komparator LM311

Na rysunku 8.2 widzimy schemat układu wykrywacza metanu.

Podstawowym elementem tworzonego układu jest **komparator LM311**. Jak sama nazwa wskazuje, komparatory służą do porównywania napięć. Jeżeli napięcie na biegunie dodatnim (+) jest wyższe od napięcia na biegunie ujemnym (-), na wyjściu komparatora popłynie prąd. W naszym przypadku zaświeci się wtedy dioda LED, a brzęczyk wyda dźwięk.



Rysunek 8.2. Schemat wykrywacza metanu

Potencjometr ustanawia napięcie graniczne na biegunie ujemnym komparatora. Aby można było korzystać z wykrywacza, rezystor ten musi zostać umieszczony w sposób uniemożliwiający świecenie diody LED. Powinna być ona włączana dopiero wtedy, gdy napięcie na wyjściu czujnika przekroczy wartość progową minusowego wejścia komparatora.

Czujnik ma niestandardowe połączenia. Wychodzi z niego sześć nóżek, ale niektóre są ze sobą połączone (rysunek 8.1). Nóżki *H* obsługują element grzewczy, który dostarcza ciepła warstwie katalizatora umieszczonej pomiędzy nóżkami *A* i *B*. Po wykryciu dowolnego stężenia metanu rezystancja pomiędzy wyjściami *A* oraz *B* maleje. Rezystor *R2* wraz z modułem czujnika tworzą dzielnik napięcia. Zaletą rozwiązania, w którym są wykorzystywane dwa rezystory — jeden jako grzałka, a drugi jako czujnik — jest możliwość dowolnej konfiguracji nóżek (można odwrotnie podłączyć układ).

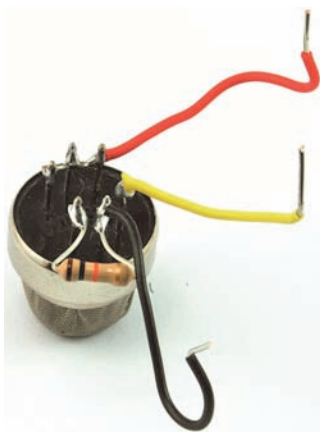
Wyjścia czujnika są dość grube oraz dziwnie rozmieszczone, nie pasują więc do płytki montażowej. Z tego powodu dolutujemy do nich kilka przewodów (rysunek 8.3).

Nie musimy lutować wszystkich wyjść, wystarczy dołączyć kable do następujących połączeń:

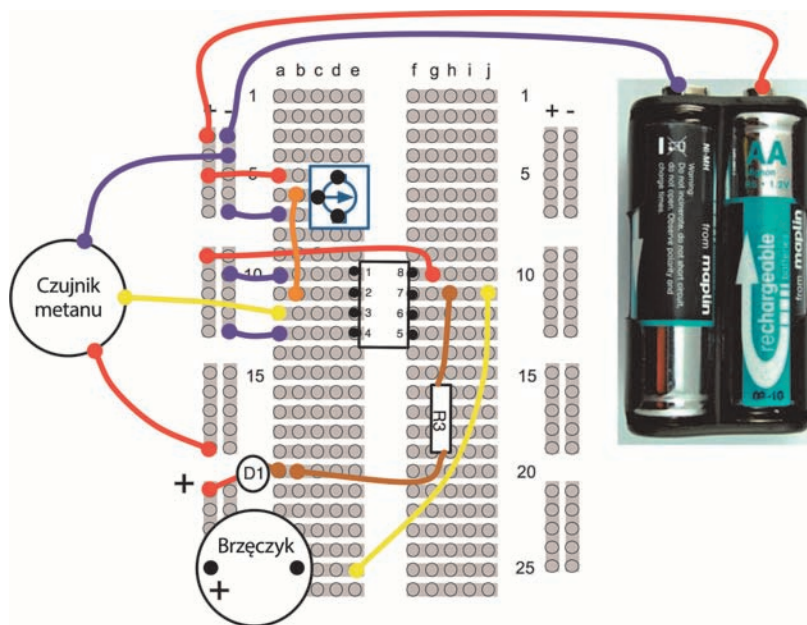
- czerwony przewód obwodu dodatniego do wszystkich nóżek z jednej strony czujnika (dwóch wyjść *A* oraz jednego *H*),
- rezystor *R2* pomiędzy nóżką *B* a masą grzałki,
- przewód masy do obwodu ujemnego grzałki (czarny),
- przewód wyjściowy do nóżki *B* (żółty).

Płytki montażowa

Rysunek 8.4 przedstawia rozmieszczenie elementów na płytce montażowej, a na rysunku 8.5 zamieszczono zdjęcie już złożonego urządzenia.

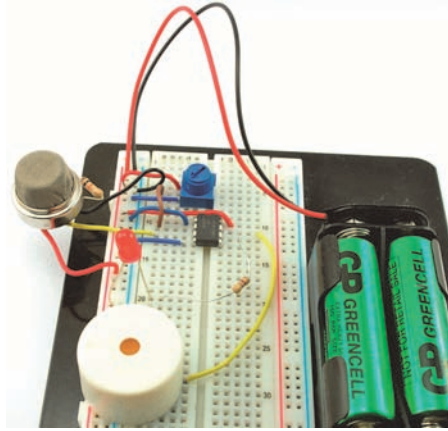


Rysunek 8.3. Podłączenie przewodów do czujnika



Rysunek 8.4. Rozmieszczenie elementów układu wykrywacza gazów na płytce montażowej

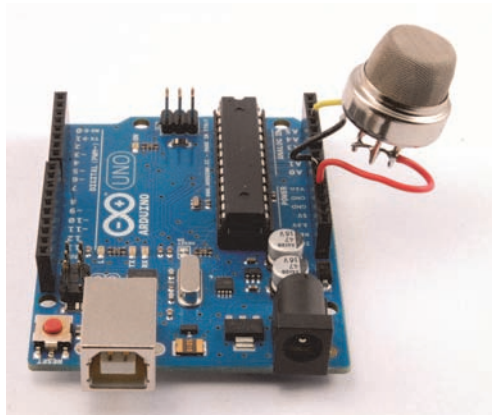
Architektura układu nie jest skomplikowana, powinieneś jednak uważać, aby komparator został właściwie zorientowany. Gdy już złożysz całość, możesz przetestować urządzenie. Wybór metody pozostawiam Tobie. Podpowiem tylko, że wydychane powietrze również uruchamia czujnik.



Rysunek 8.5. Wykrywacz gazu

Wykrywacz gazu i Arduino

Dzięki temu, że przylutowaliśmy przewody do nóżek czujnika, możemy go w prosty sposób podłączyć bezpośrednio do Arduino (rysunek 8.6).



Rysunek 8.6. Wykrywacz metanu podłączony do Arduino

Podłączmy przewód dodatni czujnika do nóżki **5V** Arduino, masę do wyjścia **GND**, a wyjście czujnika do nóżki **A3**.

Czujnik może zużywać do 200 mA prądu, musimy więc podłączyć go do właściwego zasilania 5 V oraz do masy. W tym przypadku nie przejdzie sztuczka z zasilaniem układu poprzez wyjście cyfrowe.

Kod w poniższym szkicu (*metan*) powoduje wyświetlanie odczytów czujnika w Monitorze portu szeregowego. Pamiętaj, że oddychanie w kierunku czujnika również zmienia wartości odczytów.

```
// metan

int analogPin = 3;

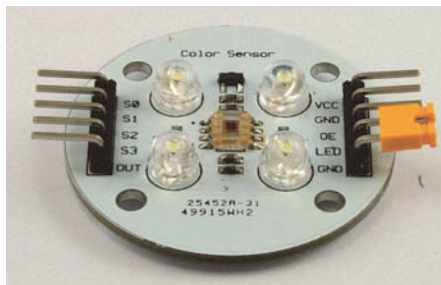
void setup()
{
  Serial.begin(9600);
  Serial.println("Wykrywacz metanu");
}

void loop()
{
  Serial.println(analogRead(analogPin));
  delay(500);
}
```

Pomiar barwy przedmiotu

Układ TCS3200 jest niewielkim, przydatnym podzespołem służącym do pomiaru barwy badanego obiektu. Istnieje kilka odmian tego układu, wszystkie działają jednak na tej samej zasadzie. Kość jest umieszczona w przezroczystej obudowie, pod którą znajdują się diody LED z nałożonymi na nie filtrami kolorów (czerwony, zielony i żółty). Dzięki temu układ może zliczać względną zawartość podstawowych kolorów w barwie danego przedmiotu.

Najprostszym sposobem wykorzystania tego układu jest stworzenie modułu takiego jak widoczny na rysunku 8.7.



Rysunek 8.7. Moduł rozpoznający kolory

Moduł ten, wart nie więcej niż 40 zł, zawiera również cztery białe diody LED służące do oświetlenia badanego obiektu, a także złączki składające się z wygodnie rozmieszczonych nóżek.

W tabeli 8.1 zostało opisane rozmieszczenie oraz przeznaczenie nóżek. Oprócz nóżki zasilającej diody LED wszystkie inne połączenia wychodzą bezpośrednio z układu scalonego, więc prawdopodobnie inne moduły zawierające kość TCS3200 będą miały te same nóżki, chociaż mogą one być nieco inaczej rozmieszczone.

Tabela 8.1. Rozmieszczenie nóżek w module czujnika kolorów

Nóżka	Opis	Opis	Nóżka
S0	Nóżki S0 i S1 regulują zakres częstotliwości. Obydwie powinny mieć ustawiony stan HIGH	Od 2,5 V do 5 V	VCC
S1		Masa	GND
S2	Czerwona — S2 i S3 w stanie LOW Zielona — S2 i S3 w stanie HIGH Niebieska — S2 w stanie LOW,	Włączenie wyjścia (ang. <i>output enable</i>) — jeżeli ustawimy stan LOW, uruchomimy układ scalony	OE
S3	S3 w stanie HIGH Biała — S2 w stanie HIGH, S3 w stanie LOW		LED
OUT	Impulsy wyjściowe		GND

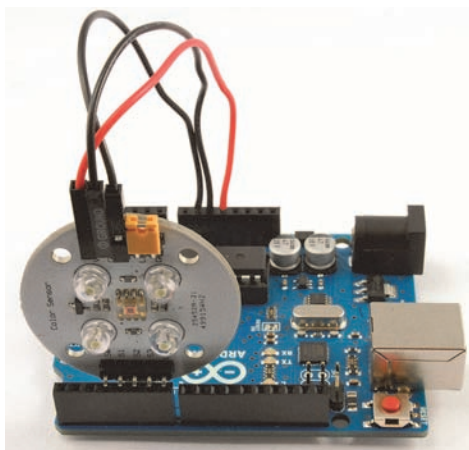
Układ scalony nie generuje żadnego sygnału na wyjściu analogowym, lecz rozróżnia częstotliwość ciągu impulsów. Za pomocą zmiany stanów cyfrowych wejść **S2** i **S3** dobieramy kolor, któremu odpowiada dana częstotliwość.

Co będzie potrzebne

Liczba	Przedmiot	Kod w dodatku
1	Arduino Uno/Leonardo	M2/M21
1	Przewód USB: typu B dla Arduino Uno, microUSB dla Arduino Leonardo	
1	Moduł czujnika kolorów	M12
1	Zestaw złązek męsko-damskich	N12

Konstrukcja

„Konstrukcja” jest w tym przypadku chyba nieco za mocnym słowem. Moduł zostanie umieszczony bezpośrednio na Arduino (rysunek 8.8), a jego przednia część będzie skierowana na zewnątrz.



Rysunek 8.8. Czujnik koloru podłączony do Arduino

Wprowadzimy następujące połączenia:

- z nóżki **S0** modułu do nóżki **D3** Arduino,
- z nóżki **S1** modułu do nóżki **D4** Arduino,
- z nóżki **S2** modułu do nóżki **D5** Arduino,
- z nóżki **S3** modułu do nóżki **D6** Arduino,
- z nóżki **OUT** modułu do nóżki **D7** Arduino.

Musimy jeszcze podłączyć trzy męsko-damskie przewody złączowe do następujących wyjść:

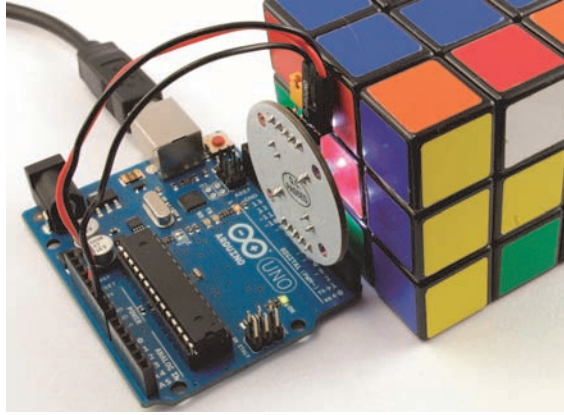
- z nóżki **VCC** modułu do nóżki **5V** Arduino,
- z nóżki **GND** modułu do nóżki **GND** Arduino,
- z nóżki **OE** modułu do nóżki **GND** Arduino.

Rysunek 8.9 przedstawia mechanizm rozpoznawania kolorów na przykładzie kostki Rubika.

Oprogramowanie

Moduł czujnika kolorów został wykorzystany w szkicu *rozpoznawanie_barw*.

```
// rozpoznawanie_barw
int pulsePin = 7;
int prescaleOPin = 3;
```



Rysunek 8.9. Rozpoznawanie kolorów na kostce Rubika

```
int prescale1Pin = 4;
int colorSelect0pin = 5;
int colorSelect1pin = 6;
```

Nazwy zmiennych identyfikujących nóżki odzwierciedlają ich przeznaczenie, a nie oznaczenie na module.

W funkcji `setup` nóżkom zostają przypisane właściwe stany oraz konfigurujemy dwie nóżki odpowiedzialne za „przeliczenie wstępne” (ang. *prescale*), które definiują zakres częstotliwości wyjściowej jako HIGH, uruchamiają komunikację szeregową oraz powodują wyświetlenie powitalnego komunikatu.

```
void setup()
{
  pinMode(prescale0Pin, OUTPUT);
  pinMode(prescale1Pin, OUTPUT);
  // określa maksymalną wartość wstępnego przeliczania
  digitalWrite(prescale0Pin, HIGH);
  digitalWrite(prescale1Pin, HIGH);
  pinMode(colorSelect0pin, OUTPUT);
  pinMode(colorSelect1pin, OUTPUT);
  pinMode(pulsePin, INPUT);
  Serial.begin(9600);
  Serial.println("Czytnik kolorow");
}
```

Funkcja `loop` odczytuje trzy różne kolory (później do tego wrócimy) oraz wyświetla komunikat w zależności od dominującej barwy. Zwróć uwagę, że im niższa wartość, tym dany kolor jest jaśniejszy.

```
void loop()
{
  long red = readRed();
```

```

long green = readGreen();
long blue = readBlue();
if (red < green && red < blue)
{
  Serial.println("CZERWONY");
}
if (green < red && green < blue)
{
  Serial.println("ZIELONY");
}
if (blue < green && blue < red)
{
  Serial.println("NIEBIESKI");
}
delay(500);
}

```

Każda z wymienionych funkcji — `readRed`, `readGreen`, `readBlue` oraz `readWhite` — wywołuje funkcję `readColor` zawierającą odpowiednie wartości nóżek **S2** i **S3**.

```

long readRed()
{
  return (readColor(LOW, LOW));
}

```

W funkcji `readColor` konfigurujemy najpierw odpowiednie nóżki dla danego koloru i w zmiennej `start` rejestrujemy czas rozpoczęcia odczytu. Następnie zbieramy odczyty z 1000 impulsów. Na koniec jest zwracana różnica pomiędzy bieżącym czasem a czasem rozpoczęcia odczytu.

```

long readColor(int bit0, int bit1)
{
  digitalWrite(colorSelect0pin, bit0);
  digitalWrite(colorSelect1pin, bit1);
  long start = millis();
  for (int i=0; i< 1000; i++)
  {
    pulseIn(pulsePin, HIGH);
  }
  return (millis() - start);
}

```

Wstawiłem również funkcję wyświetlającą wartości kolorów w Monitorze portu szeregowego, chociaż z niej nie korzystamy.

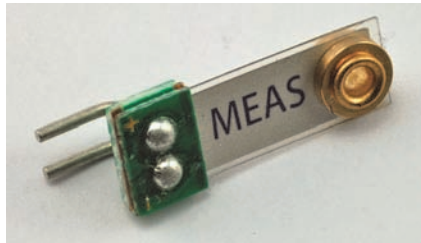
```

void printRGB()
{
  Serial.print(readRed()); Serial.print("\t");
  Serial.print(readGreen()); Serial.print("\t");
  Serial.print(readBlue()); Serial.print("\t");
  Serial.println(readWhite());
}

```

Detekcja drgań

Możemy bardzo łatwo wykorzystywać piezoelektryczne czujniki wibracji we współpracy z Arduino. Jeden z przykładowych egzemplarzy (z serwisu SparkFun) jest widoczny na rysunku 8.10.



Rysunek 8.10. Piezoelektryczny czujnik wibracji

Czujnik wibracji składa się z cienkiego paska materiału piezoelektrycznego, na którego końcu jest umieszczony nit pełniący funkcję ciężarka. Gdy pojawiają się wibracje, ciężarek porusza się i wygina płytkę piezoelektryczną, co powoduje skoki napięcia. Jeżeli mamy odpowiedni sprzęt, to napięcie może osiągnąć wartość nawet 80 V. Ponieważ jednak podłączymy czujnik do cyfrowego wejścia Arduino, jego rezystancja będzie na tyle duża, aby stłumić to napięcie do poziomu niezagrażającego elektronice mikroprocesora.

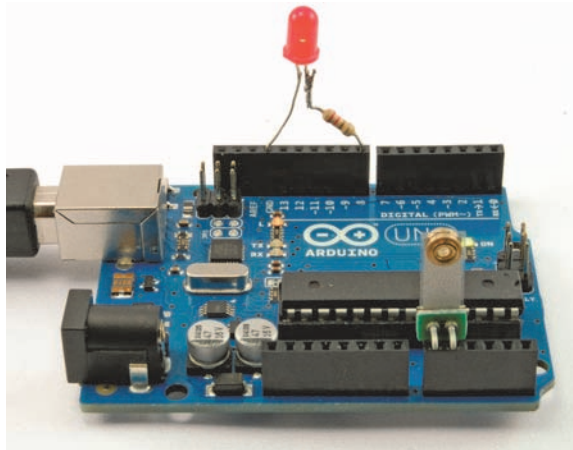
Co będzie potrzebne

Aby wykrywać drgania za pomocą czujnika piezoelektrycznego, potrzebne nam będą następujące elementy:

Liczba	Przedmiot	Kod w dodatku
1	Arduino Uno/Leonardo	M2/M21
1	Przewód USB: typu B dla Arduino Uno, microUSB dla Arduino Leonardo	
1	Piezoelektryczny czujnik drgań	M13
1	Dioda LED	Z1
1	Rezystor 220 Ω	Z2

Konstrukcja

Piezoelektryczny czujnik wibracji jest następnym z serii podzespołów przyjaznych układowi Arduino. Wystarczy podłączyć go do odpowiednich nóżek. My umieścimy go w gniazdach **A0** oraz **A1**. Nóżka **A0** zostanie skonfigurowana w trybie LOW i będzie stanowić połączenie czujnika z masą (rysunek 8.11). Zwróć uwagę, że z jednej strony moduł ten jest oznaczony symbolem **+**. Podłączmy w tym miejscu nóżkę **A1**.



Rysunek 8.11. Wykrywanie wibracji za pomocą Arduino

Diode LED łączymy z rezystorem tak, jak zostało to omówione w rozdziale 6. Tak zabezpieczoną diodę możemy złączyć biegunem dodatnim z nóżką 8., a ujemnym z **GND**.

Oprogramowanie

Poniższy kod wykorzystuje na samym początku technikę samoistnej kalibracji, aby czujnik wysłał komunikat o braku drgań. Następnie czekamy na pojawienie się wibracji o sile przekraczającej wartość progową, która spowoduje zaświecenie diody LED. Wciśnięcie przycisku *Reset* na Arduino sprawi, że czujnik znów będzie gotów do pomiaru wibracji.

```
// czujnik_wibracji

int gndPin = A0;
int sensePin = 1;
int ledPin = 8;
```


Po zdefiniowaniu używanych nóżek deklarujemy dwie zmienne. Zmienna `normalReading` jest wykorzystywana w procesie kalibracji (za chwilę go omówimy), natomiast w zmiennej `threshold` wstawiamy dopuszczalną wartość odczytu, po której przekroczeniu dioda LED zostanie włączona.

```
int normalReading = 0;
int threshold = 10;
```

W funkcji `setup` konfigurujemy tryby działania nóżek, a następnie wywołujemy funkcję `calibrate`, w przypadku braku drgań służącą do określenia odczytów czujnika.

```
void setup()
{
  pinMode(gndPin, OUTPUT);
  digitalWrite(gndPin, LOW);
  pinMode(ledPin, OUTPUT);
  normalReading = calibrate();
}
```

Funkcja `loop` po prostu odczytuje wyniki i porównuje je z wartością progową. Jeżeli ją przekroczy, dioda LED zostanie włączona.

```
void loop()
{
  int reading = analogRead(sensePin);
  if (reading > normalReading + threshold)
  {
    digitalWrite(ledPin, HIGH);
  }
}
```

Czujnik kalibrujemy za pomocą 100 impulsów wysyłanych w milisekundowych odstępach czasu. Zwracana jest wartość średnia. Wszystkie wyniki są przechowywane w zmiennej typu `long`, ponieważ mogą się one nie mieścić w standardowym typie `int`.

```
int calibrate()
{
  int n = 100;
  long total = 0;
  for (int i = 0; i < n; i++)
  {
    total = total + analogRead(sensePin);
    delay(1);
  }
  return total / n;
}
```



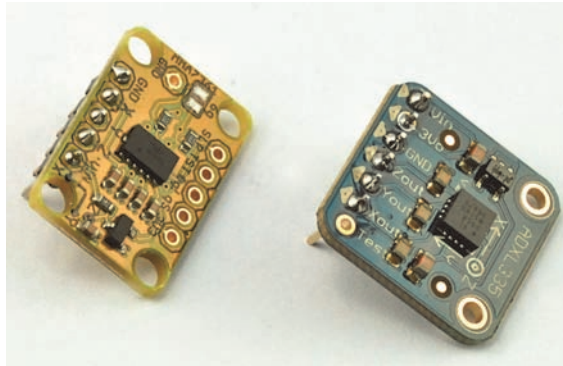
```

Serial.print(" C ");
Serial.print(tempF);
Serial.println(" F");
delay(1000);
}

```

Akcelerometr

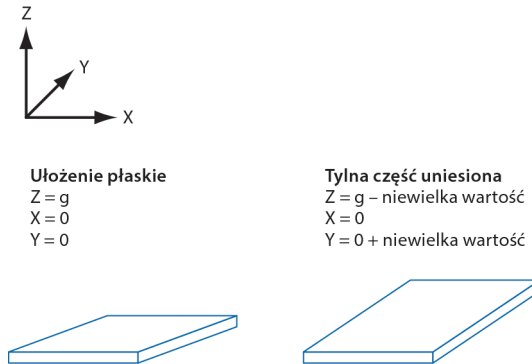
Za niewielką opłatą możesz zaopatrzyć się w małe moduły akcelerometrów (rysunek 8.14). Obydwa pokazane na rysunku modele są do siebie bardzo podobne. Są zasilane napięciem 5 V i mają wyjście cyfrowe dla każdej osi. Model widoczny z lewej strony jest dostępny na stronie Freetronics (<http://www.freetronics.com/products/3-axis-accelerometer-module-.Ug4yCpL0FT4>), natomiast moduł po prawej możesz zakupić w serwisie Adafruit (<http://www.adafruit.com/products/163>).



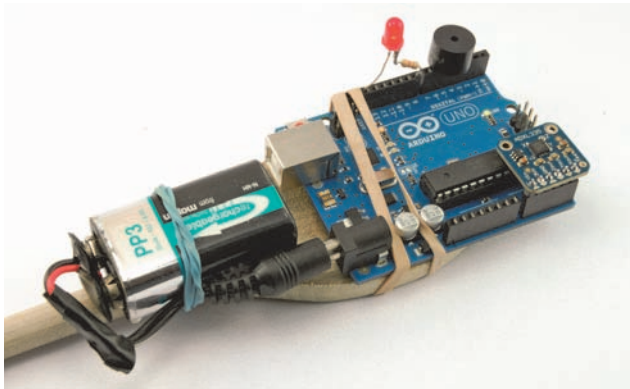
Rysunek 8.14. Moduły akcelerometrów

Moduły te są zbudowane z akcelerometrów trójosiowych, mierzących siłę działającą na niewielki ciężarek umieszczony wewnątrz układu. Dwie osie (X i Y) są ułożone równolegle do płytki drukowanej. Oś Z jest prostopadła w stosunku do układu scalonego. Pod wpływem grawitacji w tym kierunku będzie oddziaływała siła. Jeśli więc przechylimy nieco moduł, zmieni się wartość oddziaływania grawitacyjnego po unoszonej stronie (rysunek 8.15).

Przetestujemy jeden z omawianych akcelerometrów na elektronicznej wersji wyścigu z jajkiem na łyżce. Wykorzystamy nasz moduł do wykrywania poziomego przechylenia „łyżki”; w momencie gdy „jajko” będzie zagrożone wypadnięciem, zostanie zapalona dioda LED. Brzęczyk będzie nam sygnalizował przechylenie oznaczające wypadnięcie jajka (rysunek 8.16).



Rysunek 8.15. Wpływ grawitacji na akcelerometr



Rysunek 8.16. Wyścig z Arduino na łyżce

Co będzie potrzebne

Aby wziąć udział w wyścigu z Arduino na łyżce, będą nam potrzebne następujące podzespoły:

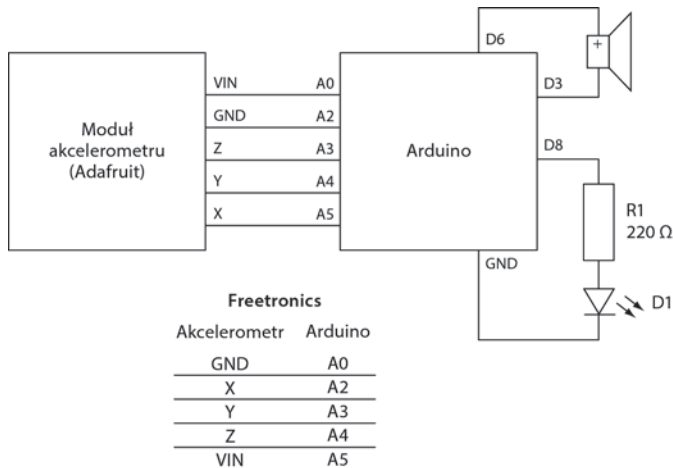
Liczba	Przedmiot	Kod w dodatku
1	Arduino Uno/Leonardo	M2/M21
1	Przewód USB: typu B dla Arduino Uno, microUSB dla Arduino Leonardo	
1	Akcelerometr	M15
1	Brzęczyk piezoelektryczny	M3
1	Dioda LED	Z1

Liczba	Przedmiot	Kod w dodatku
1	Rezystor 220 Ω	Z2
1	Prześciówka pomiędzy zaciskiem do baterii a wtyczką zasilania 2,1 mm	S9
1	Drewniana łyżka	
1	Bateria PP3 9 V	

Konstrukcja

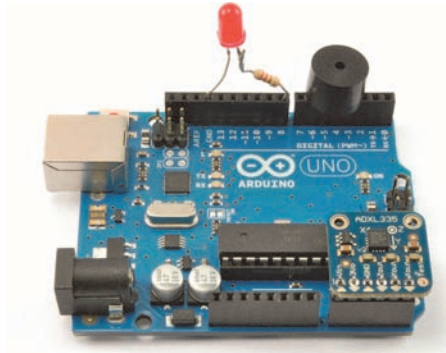
Przy odrobinie sprytu możemy jednocześnie podłączyć do Arduino zarówno akcelerometr, jak również brzęczyk i diodę LED. Należy najpierw przesłać odpowiedni szkic do mikrokontrolera przed podłączeniem modułu, gdyż w poprzednio załadowanym projekcie niektóre z nóżek od **A0** do **A5** mogły zostać skonfigurowane jako wyjścia.

Schemat naszego czujnika wykorzystywanego w wyścigu został umieszczony na rysunku 8.17.



Rysunek 8.17. Schemat układu mierzącego przechylenie łyżki

Jak widać na rysunku 8.18, wszystkie elementy mają swoje miejsce w gniazdach Arduino. Połączyliśmy diodę LED z rezystorem zgodnie z opisem umieszczonym w rozdziale 6. Biegun dodatni diody łączy się z nóżką 8. Arduino, a biegun ujemny — z **GND**. Brzęczyk umieściliśmy pomiędzy nóżkami **D3** i **D6** — ta druga łączy się z biegunem dodatnim brzęczyka. Jeżeli nóżki brzęczyka są rozmieszczone w inny sposób, możesz wybrać inne wejścia w Arduino, pamiętaj jednak, żeby odpowiednio zmienić również wartości zmiennych `gndPin2` i `buzzerPin` w kodzie.



Rysunek 8.18. Arduino wraz z podłączonymi podzespołami

Jak pokazano na rysunku 8.18, obydwa moduły akcelerometrów mieszczą się w gniazdach od *A0* do *A5*, jednak różnią się one rozmieszczeniem nóżek.

Układ jest zasilany baterią 9 V podłączoną za pomocą przejściówki, a całość została umocowana na łyżce przy użyciu gumek recepturek.

Oprogramowanie

W tym projekcie są wykorzystywane dwie wersje szkicu: *jajko_na_lyzce_adafruit* oraz *jajko_na_lyzce_freetronics*. Wybierz właściwy szkic, a następnie zaprogramuj Arduino, **ZANIM** podłączysz moduł akcelerometru.

Jedyna różnica pomiędzy tymi szkicami polega na rozmieszczeniu nóżek.

Opiszę tutaj wersję dla akcelerometru Adafruit.

Rozpoczynamy od zdefiniowania używanych nóżek.

```
//jajko_i_lyzka_adafruit
```

```
int gndPin1 = A2;
int gndPin2 = 3;
int xPin = 5;
int yPin = 4;
int zPin = 3;
int plusPin = A0;
int ledPin = 8;
int buzzerPin = 6;
```

Jeśli łyżka jest ustawiona poziomo, zmienne `levelX` oraz `levelY` służą do określania wartości spoczynkowej przyspieszenia dla osi X i Y.

```
int levelX = 0;
int levelY = 0;
```

Dzięki zmiennym `ledThreshold` oraz `buzzerThreshold` możemy modyfikować kąt przechylenia łyżki, przy którym dioda LED zaczyna świecić, a brzęczyk generuje dźwięk ostrzegający o „wypadnięciu jajka”.

```
int ledThreshold = 10;
int buzzerThreshold = 40;
```

Funkcja `setup` uaktywnia nóżki, a następnie wywołuje funkcję `calibrate`, służącą do ustawiania wartości zmiennych `levelX` oraz `levelY`.

```
void setup()
{
  pinMode(gndPin1, OUTPUT);
  digitalWrite(gndPin1, LOW);
  pinMode(gndPin2, OUTPUT);
  digitalWrite(gndPin2, LOW);
  pinMode(plusPin, OUTPUT);
  pinMode(ledPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
  digitalWrite(plusPin, HIGH);
  pinMode(A1, INPUT); // wyjście 3 V
  calibrate();
}
```

W głównej pętli odczytujemy wartości przyspieszenia w osiach X i Y oraz sprawdzamy, jak bardzo oddaliły się one od wartości zmiennych `levelX` i `levelY`. Funkcja `abs` zwraca wartość bezwzględną wyniku, więc jeśli odczyt będzie liczbą ujemną, zostanie zmieniony na liczbę dodatnią, która dopiero teraz może zostać przyrównana do wartości progowych.

```
void loop()
{
  int x = analogRead(xPin);
  int y = analogRead(yPin);
  boolean shakey = (abs(x - levelX) > ledThreshold || abs(y - levelY) > ledThreshold);
  digitalWrite(ledPin, shakey);
  boolean lost = (x > levelX + buzzerThreshold || y > levelY + buzzerThreshold);
  if (lost)
  {
    tone(buzzerPin, 400);
  }
}
```

Jedynym minusem funkcji `calibrate` jest konieczność odczekania 200 milisekund, zanim pojawią się pierwsze odczyty. Akcelerometr zostaje w tym czasie uruchomiony.

```
void calibrate()
{
  delay(200); // dajemy akcelerometrowi czas na włączenie się
  levelX = analogRead(xPin);
  levelY = analogRead(yPin);
}
```


Wykrywanie pól magnetycznych

Wykrywanie pola magnetycznego staje się prostą czynnością dzięki wykorzystaniu takiego trójwyciowego podzespołu jak na przykład liniowy czujnik efektu Halla A1302. Możemy używać tego układu tak samo, jak wykorzystywaliśmy czujnik temperatury TMP36 w podrozdziale „Pomiar temperatury”.

Co będzie potrzebne

Podczas stosowania modułu pomiaru pola magnetycznego wykorzystamy następujące podzespoły:

Liczba	Przedmiot	Kod w dodatku
1	Arduino Uno/Leonardo	M2/M21
1	Przewód USB: typu B dla Arduino Uno, microUSB dla Arduino Leonardo	
1	Liniowy czujnik efektu Halla A1302	P12

Konstrukcja

Tak samo jak układ TMP36, czujnik A1302 ma tylko trzy nóżki; dwie służą do zasilania, a trzecia stanowi wyjście analogowe. Napięcie zasilające mieści się w zakresie od 4,5 V do 6 V, dzięki czemu zasilanie 5 V Arduino całkowicie zaspokoi nasze potrzeby energetyczne.

W rzeczywistości zupełnie wystarczające okazuje się zasilanie z wyjść cyfrowych, możemy więc umieścić wszystkie trzy nóżki czujnika w złączu analogowym Arduino (rysunek 8.19). Kółko na czujniku powinno być skierowane na zewnątrz.

Nóżka **A1** Arduino może być zaprogramowana jako wyjście, dlatego przed podłączeniem czujnika należy najpierw wczytać szkic do mikrokontrolera.

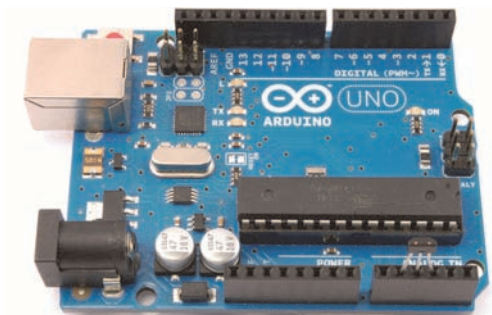
Oprogramowanie

Kod dla czujnika magnetycznego bardzo przypomina szkic termometru.

Deklarujemy najpierw trzy nóżki: cyfrowe wejścia 15. i 17. (nóżki **A0** i **A2**), natomiast **A1** definiujemy jako nóżkę czujnika.

```
// czujnik_magnetyczny

int gndPin = A1;
int sensePin = 2;
int plusPin = A3;
```



Rysunek 8.19. Czujnik magnetyczny A1302 umieszczony w Arduino

```
void setup()
{
  pinMode(gndPin, OUTPUT);
  digitalWrite(gndPin, LOW);
  pinMode(plusPin, OUTPUT);
  digitalWrite(plusPin, HIGH);
  Serial.begin(9600);
}
```

Główna pętla pobiera jedynie nieprzetworzone odczyty i wyświetla je w Monitorze portu szeregowego.

Urządzenie nie jest zbyt czułe, jednak jeśli zbliżymy do niego magnes, wartości wyświetlane w Monitorze portu szeregowego powinny ulec zmianie.

```
void loop()
{
  int raw = analogRead(sensePin);
  Serial.println(raw);
  delay(1000);
}
```

Podsumowanie

Istnieje mnóstwo rodzajów czujników, które nie zostały omówione, i wiele z nich można bez większego problemu podłączyć do Arduino, czy to przez wejście analogowe, czy też za pomocą modulacji impulsów. Szkice opisane w tym rozdziale można z łatwością dostosować do korzystania z innych czujników.

W następnym rozdziale zmienimy nieco kierunek eksperymentów i zajmiemy się elektroniką związaną z dźwiękiem i muzyką.

Skorowidz

A

AC, *Patrz:* prąd zmienny
adres
 IP, 177, 179
 MAC, 177, 179
akcelerometr, 256, 258, 281, 282
 trójosiowy, 256
akumulator, 121, 125, *Patrz też:* bateria
 do telefonu komórkowego, 133
 kwasowo-ołowiowy, 127, 130, 131
 litowo-polimerowy, 125, 132, 133
 ładowanie, 126, 127, 129, 130, 131, 132
 NiMH, 129, 130
 pojemność, 128
 przeciążenie, 128
 żywość, 128
alarm przeciwwłamaniowy, 196
Allen Charlie, 187
alternating current, *Patrz:* prąd zmienny
amper, 53
amperogodzina, 122
anoda, 47, 48, 88, 91, 167, 298
 wspólna, 232
Arduino, 147
 konfiguracja, 149
 Leonardo, 148

moduł rozszerzeń, 173
nóżka, 155
program, *Patrz:* szkic
Uno, 148
zasilanie, 176, 213

B

bateria, 45, 50, 58, 121, *Patrz też:* akumulator
 bezpieczna wartość rozładowania, 122
 jednorazowa, 122
 ładowanie, 127
 pojemność, 121, 128
 jednostka, 122
 przeciążenie, 128
 rezystancja wewnętrzna, 122
 testowanie, 291
bezpiecznik, 289
 prąd znamionowy, 289
 szybki, 289
 zwłoczny, 290
biblioteka
 Adafruit, 234
 Ethernet, 179
 SPI, 179

bramka, 76
buck-boost converter, *Patrz:* przekształtnik
obniżająco-podwyższający

C

capacity, *Patrz:* pojemność
cewka, 274
charliepleksing, 187
chip, *Patrz:* układ scalony
cisza, 268
Continuity test, *Patrz:* test ciągłości obwodu
cyna lutownicza, *Patrz:* lut
częstotliwość radiowa, 207
czujnik, 241
 efektu Halla, 261
 kolorów, 246, 248
 metanu, 241
 piezoelektryczny, 251, 252
 podczerwieni, 198
 pola magnetycznego, 261
 ruchu, 198
 PIR, 196
 temperatury, 254
 wibracji, 251, 252

D

dalmierz, 200, 202, 205
DC, *Patrz:* prąd stały
DC Current, *Patrz:* natężenie prądu stałego
DC Volts, *Patrz:* napięcie prądu stałego
dioda, 47, 57, 88, 140, 294
 anoda, *Patrz:* anoda
 katoda, *Patrz:* katoda
 laserowa, 114
 napięcie
 przewodzenia, 89, 91, 298
 zaporowe, 90
 prąd przewodzenia, 89
 wartość znamionowa, 90
 spolaryzowana, 88, 89
 świecąca, *Patrz:* LED
 testowanie, 298

direct current, *Patrz:* prąd stały
drain, *Patrz:* dren
dren, 76
drut połączeniowy, *Patrz:* przewód
 jednożyłowy
dzielnik napięcia, 61, 163
 obliczanie, 64
dźwięk, *Patrz:* sygnał dźwiękowy

E

EAGLE PCB, 307
element
 elektroniczny
 do montażu powierzchniowego,
 Patrz: SMD
 odzyskiwanie, 294
 opornik, *Patrz:* rezystor
 oznaczenia kodowe, 22, 44, 45
 rezystor, *Patrz:* rezystor
 symbole, 57
 zakup, 22
 źródła, 22, 23
 grzewczy, 241, 243
 testowanie, 291
 katalityczny, 241
energia elektryczna, 54
eterofon, 279

F

farad, 45
fotorezystor, 64, 67, 69, 279
fototranzystor, 50
Fritzing, 306

G

gate, *Patrz:* bramka
generator
 dźwięku, 170
 sygnałów, 279

głośnik, 61, 294
 budowa, 274
 moc, 275
 niskotonowy, 275
 przestrzenny, 275
 rezystancja, 274
 wysokotonowy, 275
 zakres częstotliwości, 275
 GND, *Patrz:* masa

H

hookup wire, *Patrz:* przewód jednożyłowy

I

IDE, 147
 Integrated Development Environment,
Patrz: IDE
 interfejs szeregowy, *Patrz:* magistrala

J

jednostka
 napięcia, 53
 natężenia prądu, 53
 pojemności, 45
 rezystancji, 43
 jumper wire, *Patrz:* przewód z wtyczkami
 na końcach

K

kabel, 26, 28
 kalkulator sieciowy, 307
 kandela, 93
 katalizator, 243
 katoda, 47, 48, 88
 wspólna, 232, 233
 klawiatura
 emulowanie, 281
 USB, 192
 koder obrotowy, 50
 komparator, 242, 244, 270

kondensator, 45, 57, 116, 135, 287
 eksplozja, 47
 elektrolityczny, 46, 299
 ładowanie, 45
 nietypowy, 294
 odsprzęgający, 45
 pojemność, *Patrz:* pojemność
 polaryzacja, 46
 pomiar, 298
 SMD, 50
 spolaryzowany, 57
 sprzęgający, 276
 testowanie, 292
 wygładzający, 45
 kontroler muzyczny, 281, 283
 kość, *Patrz:* układ scalony

L

LED, 47, 58, 67, 87, 166, 167, 196, 232
 czerwona, 89, 246
 częstotliwość migotania, 166
 dużej mocy, 95
 jasność, 169
 kalkulator, 102
 karta katalogowa, 90
 kąta emisji, 93
 kolor, 91, 93
 łączenie szeregowo, 102
 macierz, 187
 migotanie, 168
 napięcie przewodzenia, 100
 podczerwone, 94
 RGB, 48, 94
 spolaryzowana w kierunku przewodzenia, 89
 światłość, 93
 ultrafioletowe, 94
 zielona, 246
 żółta, 246
 light emitting diode, *Patrz:* LED
 lut, 31, 32
 opary, 35

lutowanie, 28, 29, 32, *Patrz też:* wylutowywanie przewodów dźwiękowych, 265, 266 zasady bezpieczeństwa, 29, 38
lutownica, 23
czyszczenie, 31

Ł

ładowanie podtrzymujące, 128, 133, 141, 144
ładowarka, 128
 równoważona, 132
 telefoniczna, 295
ładunek elektryczny, 45
łazik elektroniczny, 226

M

magic hands, *Patrz:* trzecia ręka
magistrala I2C, 233, 234
magnes, 274
masa, 36
materiał piezoelektryczny, 251
Metal Oxide Semiconductor Field Effect Transistor, *Patrz:* tranzystor MOSFET
miernik, 297
 głośności, 284
 tryb testowania ciągłości, *Patrz:* test ciągłości obwodu
uniwersalny, 23
zakres
 stałoprądowy, 37, 62
 zmiennoprądowy, 37
mikrofarad, 46
mikrofon, 268
mikrokontroler, 49, 82, 147
 Arduino, *Patrz:* Arduino
 zmiana trybu działania nóżek, 187
mikroprzełącznik, 83
miliamper, 53
milikandela, 93
moc, 54
 jednostka, 54
 znamionowa, 61

modulacja szerokości impulsów, *Patrz:* PWM
moduł
 Ethernetowy, 148, 174, 179
 laserowy, 114
 mikrofonowy, 268, 284
 odbiornika, 226
 rozszerzeń, 147
 silnika, 226
 zegara RTC, *Patrz:* zegar RTC
monitor portu szeregowego, 160, 196, 254
mostek H, 215, 217, 241
multipleksing, 233

N

nadajnik radiowy, 271
nanofarad, 46
napięcie, 51, 52, 54, 301
 dzielnik, *Patrz:* dzielnik napięcia
 jednostka, 53
 odniesienia, 36
 polaryzacja, 47
 pomiar, 100
 porównywanie, 242
 prądu stałego, 24
regulator, 135
 LDO, 137
skuteczne, 300
tolerancja, 135
znamionowe, 46
narzędzia, 23
natężenie, 51, 301
 jednostka, 53
 prądu stałego, 24, 44
nożyce, 23
 do cięcia drutu, 27

O

obwód
 dodatni, *Patrz:* obwód prądowy
 drukowany, 50
 niestabilność, 45

prądowy, 36, 46
 ujemny, *Patrz:* masa
 zwarcie, 216
 odbiornik radiowy, 61
 ogniwo AA, 121
 ogniwo słoneczne, 142
 Ohma prawo, *Patrz:* prawo Ohma
 om, 53
 OmniGraffle, 107
 opornik, *Patrz:* rezystor
 oporność, *Patrz:* rezystancja
 oprogramowanie, 305
 EAGLE PCB, 307
 Fritzing, 306
 symulator obwodów elektronicznych, 306
 oscyloskop, 268, 304

P

pamięć, 49
 panel słoneczny, 121, *Patrz:* ogniwo słoneczne
 Passive Infra Red, *Patrz:* podczerwień
 pasywna
 pikofarad, 45, 46
 pilot bezprzewodowy, 207, 208, 209
 pluskwa radiowa, 271
 płytki
 drukowana, 34
 montażowa, 24, 62
 perforowana, 106, 109, 110
 prototypowa, 50
 podczerwień pasywna, 196
 podzespół do montażu powierzchniowego, 50,
Patrz: SMD
 pojemność, 45, 128
 pomiar, 298
 pole magnetyczne, 261
 połączenie suche, 34
 pomiar
 napięcia prądu stałego, 24
 natężenia prądu stałego, 24
 pojemności, 298
 rezystancji, 24

port
 COM, 150
 szeregowy, 160
 potencjometr, 45, 50, 62, 163
 prawo Ohma, 52, 69
 prąd
 napięcie, *Patrz:* napięcie
 natężenie, *Patrz:* natężenie
 rezystancja, *Patrz:* rezystancja
 stały, 37, 301
 sterownik, *Patrz:* sterownik
 prądu stałego
 zmienny, 37, 300, 302
 napięcie skuteczne, 300
 projektowanie płytek drukowanych, 307
 protokół DHCP, 177
 przedwzmacniacz mikrofonowy, 269
 przeglądarka, 177
 przekaźnik, 154, 159, 173, 208
 przekształtnik obniżająco-podwyższający, 137
 przełącznik, 37, 50, 58, 76, 81, 159, 215, 294
 chwilowy, 83
 DPDT, 84
 DPST, 83
 dwupozycyjny, 83
 dwustabilny, 83
 ON-Off-Mom, 85
 SPDT, 83
 SPST, 83
 wciskany, *Patrz:* przycisk
 przetwornik ultradźwiękowy, 201
 przewód
 dźwiękowy, 263, 264
 lutowanie, 265, 266
 wtyczka, 264, 266
 ekranowany, 27, 264
 głośnikowy, 265
 jednożyłowy, 24, 26, 28
 łączenie
 przez lutowanie, 28, 29, 31, 32
 przez skręcanie, 28
 prądowy, 37
 usuwanie izolacji, 25, 27

przewód
 wielożyłowy, 26, *Patrz:* kabel
 z wtyczkami na końcach, 24
 przycisk, 82
 PWM, 169

R

radiator, 96
 Real Time Clock, *Patrz:* zegar RTC
 Resistance, *Patrz:* rezystancja
 rezystancja, 24, 44, 51, 54, 274
 jednostka, 53
 pomiar, 298
 rezystor, 41, 43, 47, 57
 dużej mocy, 43
 moc znamionowa, 44
 nastawny, *Patrz:* potencjometr
 oznaczenia kodowe, 44, 45
 rozgrzewanie, 59
 SMD, 50
 stały, 45
 światłoczuły, 50, *Patrz:* fotorezystor
 testowanie, 292
 zmienny, *Patrz:* potencjometr
 robot Magician, 227
 router, 177

S

samochód zdalnie sterowany, 115, 117
 schemat, 61
 czytanie, 36, 55, 56
 serwer sieciowy, 177
 serwowmotor, 185, 186
 shield, *Patrz:* moduł rozszerzeń
 silnik, 229, 294
 krokowy, 221, 222
 regulacja obrotów, 212
 wirnik, 221
 zmiana kierunku obrotów, 215
 silnik elektryczny, 79
 sketch, *Patrz:* szkic

słuchawki, 27
 SMD, 50, 195
 sonar, 201
 sprzężenie zwrotne, 270
 starter kit, *Patrz:* zestaw startowy
 statyw, 23
 sterownik
 prądu stałego, 96
 przekaźnika, 173
 stripboard, *Patrz:* płytko perforowana
 Surface Mount Device, 50
 switch, *Patrz:* przełącznik
 sygnał dźwiękowy, 263, 264
 stereofoniczny, 266
 zniekształcenie, 269
 symulator obwodów elektronicznych, 306
 system przeciwkradzieżowy, 196
 szczypce, 23, 27
 szkic, 150
 szum elektryczny, 27, 45, 264

Ś

ścieżka dźwiękowa, 27
 śrubokręt, 23
 światła natężenie, *Patrz:* światłość
 światłość, 93

T

technologia typu D, 276
 temperatura, 299
 termoogniwo, 299
 test
 ciągłości obwodu, 24, 33, 272, 298
 płytki drukowanej, 34
 połączenia, 33
 timer, *Patrz:* układ czasowy 555
 tranzystor, 49, 68, 75, 215, 294
 baza, *Patrz:* baza
 bipolarny, 57, 68, 76, 78
 testowanie, 292, 302
 bramka, *Patrz:* bramka

emiter, *Patrz:* emiter
 karta katalogowa, 75
 kolektor, *Patrz:* kolektor
 MOSFET, 76, 78, 212
 zasilanie, 213
 NPN, 57, 77, 78
 PNP, 77, 78
 polowy o strukturze metal-tlenek-
 półprzewodnik, *Patrz:* tranzystor
 MOSFET
 SMD, 50
 sterowany stanem logicznym, 81
 współczynnik wzmocnienia, 302
 wzmocnienie, 68
 prądowe, 76
 z kanałem
 typu N, 78
 typu P, 78
 źródło, *Patrz:* źródło
 tranzystor NPN, 68
 trickle charging, *Patrz:* ładowanie
 podtrzymujące
 trzecia ręka, 30
 tweeter, *Patrz:* głośnik wysokotonowy

U

układ
 czasowy 555, 103, 105, 279
 scalony, 49
 7805, 135
 LM317, 96
 MCP73831, 132
 mostek H, 217
 SMD, 50
 sterownika, 182
 wspólnej anody, 232
 wspólnej katody, 232, 233
 wzmacniający, *Patrz:* wzmacniacz
 ultradźwięki, 200, 201

V

VeroDes, 107
 Volume Unit, *Patrz:* miernik głośności
 VU, *Patrz:* miernik głośności

W

wat, 54
 wolt, 53
 woltomierz, 62
 woofer, *Patrz:* głośnik niskotonowy
 współczynnik wzmocnienia, 270
 wylutowywanie, 293
 warkocz, 293
 wyświetlacz
 LCD, 182
 LED, 232, 294
 siedmiosegmentowy, 48, 232, 294
 słupkowy, 48
 trzykrotny, 232
 wzmacniacz, 268
 dźwięku, 49, 275
 operacyjny, 270

Z

zasady bezpieczeństwa, 287, 288, 289, 301, 302
 zasilacz, 121, 303
 impulsowy, 287
 zasilanie awaryjne, 139
 zegar
 cyfrowy, 237
 RTC, 236
 zestaw
 rezystorów, 41
 startowy, 41, 42
 zintegrowane środowisko programistyczne,
 Patrz: IDE
 złącze, 294
 zwrotnica, 275

Ż

- źródło, 76
 - prądu programowalne, 131
 - zasilania, 121
 - dodatnie, 26
 - ujemne, 26

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



- 1. ZAREJESTRUJ SIĘ**
- 2. PREZENTUJ KSIĄŻKI**
- 3. ZBIERAJ PROWIZJĘ**

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

Poznaj prawa rządzące światem elektroniki

— zbuduj swój pierwszy układ elektroniczny!

Fascynuje Cię elektronika? Chciałbyś zrealizować swój pierwszy projekt, ale nie wiesz, od czego zacząć? Nie martw się! Oddajemy w Twoje ręce podręcznik, dzięki któremu już wkrótce samodzielnie zaprojektujesz i zbudujesz układ elektroniczny. Zaczynaj od czegoś np. praktycznego termometru, lub czegoś całkowicie zwiariowanego, np. zestawu migających diod. Wszystko w Twoich rękach!

Jeszcze nigdy na rynku wydawniczym nie było książki tak potrzebnej osobom rozpoczynającym wędrówkę po świecie elektroniki. W trakcie lektury nauczysz się niezbędnych podstaw — lutowania, zdejmowania izolacji oraz rozpoznawania podstawowych układów elektronicznych. Każdy rozdział to kolejna dawka solidnej wiedzy: stosowanie baterii i zapewnianie zasilania, wykorzystanie diod i wiele innych nowych tematów. Rozpocznesz również przygodę z platformą Arduino. Dzięki licznym modułom daje ona ogromne pole do popisu. Dalmierze, wykrywacze ruchu, sterowanie silnikami to tylko część projektów, które zrealizujesz z Arduino. Książka ta jest doskonałym źródłem informacji dla wszystkich amatorów elektroniki!

Dzięki tej książce:

- nauczysz się czytać schematy elektroniczne
- zidentyfikujesz podstawowe elementy elektroniczne
- poznasz podstawowe prawa świata elektroniki
- wykorzystasz platformę Arduino
- zbudujesz swój pierwszy zaawansowany układ elektroniczny



helion.pl
księgarnia
internetowa

Nr katalogowy: 16985

Księgarnia internetowa:
<http://helion.pl>

Zamówienia telefoniczne:
0 801 339900
0 601 339900

 **Helion**

Sprawdź najnowsze promocje:
• <http://helion.pl/promocje>
Książki najchętniej czytane:
• <http://helion.pl/bestsellery>
Zamów informacje o nowościach:
• <http://helion.pl/nowosci>

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
<http://helion.pl>

sięgnij po WIĘCEJ



KOD KORZYŚCI

ISBN 978-83-246-8051-1



9 788324 680511

Cena: 59,00 zł

Informatyka w najlepszym wydaniu