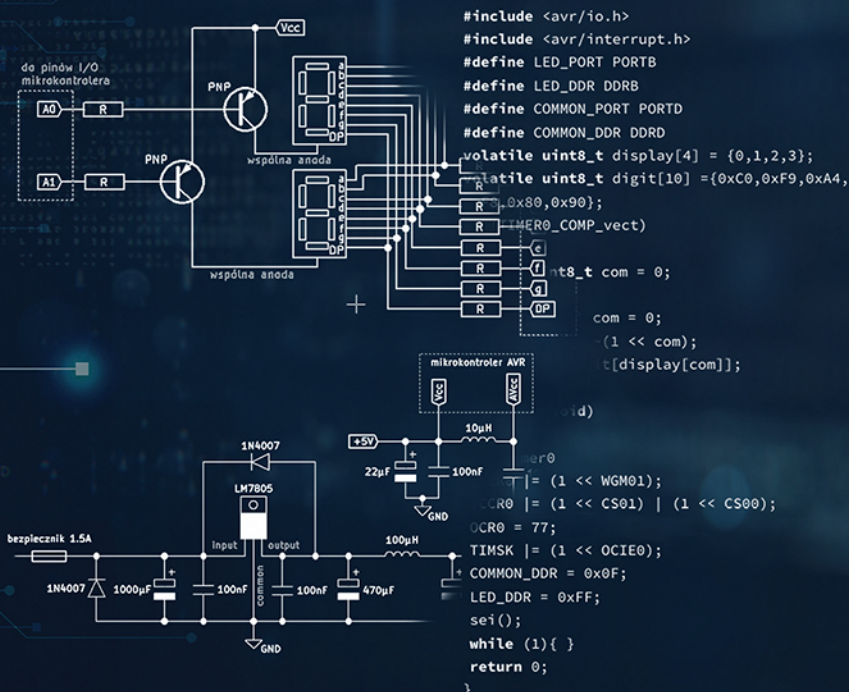


MIKROKONTROLERÓW

AVR

OD ELEKTRONIKI DO PROGRAMOWANIA



FILIP SALA
MARZENA SALA-TEFELSKA

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 Helion S.A. 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 Helion S.A. nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Szymon Sz wajger
Projekt okładki: Jan Paluch
Grafika na okładce: Filip Sala, Marzena Sala-Tefelska

Helion S.A.
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/wpravr>
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Materiały do książki znaleźć można pod adresem:
<https://ftp.helion.pl/przyklady/wpravr.zip>

ISBN: 978-83-283-7625-0

Copyright © Helion S.A. 2021

Printed in Poland.

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

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

Spis treści

Słowo wstępne	9
1 Wprowadzenie	13
1.1 Czym są mikrokontrolery AVR?	13
1.2 Jak to wszystko działa?	14
1.3 Bezpieczeństwo	16
1.4 Często zadawane pytania	22
2 Podstawy elektroniki	26
2.1 Prawo Ohma	26
2.2 Pierwsze prawo Kirchhoffa	27
2.3 Moc	28
2.4 Elementy elektroniczne	30
2.4.1 Rezystory	30
2.4.2 Potencjometry	38
2.4.3 Elementy indukcyjne	39
2.4.4 Kondensatory	43
2.4.5 Diody	57
2.4.6 Diody świecące (LED)	61
2.4.7 Tranzystory	64
2.4.8 Układy scalone	69
2.4.9 Przekazniki	74
2.4.10 Termistory	77
2.4.11 Silniki krokowe	80
2.4.12 Przełączniki	88
2.4.13 Prototypowe płytki uniwersalne	91
2.4.14 Prototypowe płytki stykowe	92
2.4.15 Obudowy elementów elektronicznych	94
2.5 Jak czytać schematy elektroniczne	96
2.6 Jak czytać noty katalogowe	100
2.7 Dobór elementów elektronicznych	102
2.8 Lutowanie i rozlutowywanie	108
2.9 Podstawowe pomiary multimetrem	117
2.9.1 Zakres pomiarowy	118

2.9.2	Podłączenie przewodów	119
2.9.3	Napięcie	121
2.9.4	Natężenie	122
2.9.5	Rezystancja, pojemność, indukcyjność	122
2.9.6	Pomiar temperatury za pomocą termopary	123
2.9.7	Pomiar ciągłości obwodu	124
2.9.8	Odczytywanie wartości	125
2.9.9	Błędy pomiarowe	129
2.10	Analizator stanów logicznych	132
2.11	Oprogramowanie wspomagające projektowanie	133
3	Elektronika mikrokontrolerów	139
3.1	Poziomy napięć i wydajność prądowa	139
3.2	Taktowanie	140
3.3	Zasilanie	143
3.4	Podłączenie programatora i zasilania	152
3.5	Reset	155
3.6	Płytki testowa	155
3.7	Porady praktyczne	156
4	Programowanie	162
4.1	Środowisko programistyczne	162
4.1.1	WinAVR i Makefile	163
4.1.2	Kompilator avr-gcc	166
4.1.3	Edytory kodu	172
4.1.4	Atmel Studio (dawniej AVR Studio)	173
4.1.5	Biblioteki avr-libc	174
4.2	Programatory	175
4.3	AVRDUDE	177
4.4	JTAG	180
4.5	Fusebity i lockbity	181
4.6	Typy danych	186
4.6.1	Typy całkowite	187
4.6.2	Typy zmiennoprzecinkowe float oraz double	189
4.6.3	Łańcuchy (<i>strings</i>)	190

4.6.4	Konwersja typów	194
4.7	Kod uzupełnień do dwóch	199
4.8	Obszary pamięci w mikrokontrolerach	201
4.8.1	Pamięć programu (flash)	202
4.8.2	Pamięć danych	203
4.8.3	EEPROM	208
4.9	System binarny	208
4.10	System heksadecymalny	209
4.11	Przesunięcia bitowe	211
4.12	Operatory logiczne i bitowe. Maskowanie	213
4.13	Priorytet operatorów w języku C	216
4.14	Rejestry	218
4.15	Pierwszy program	226
4.16	Sterowanie portami wejścia-wyjścia I/O	233
4.17	Obsługa przycisków. Drgania styków	239
4.18	Opóźnienia (delay)	254
4.19	Tworzenie własnych bibliotek	256
4.20	Nazwy funkcji bibliotek i zmiennych	261
4.21	Przerwania	264
4.22	Liczniki	276
4.22.1	Tryb porównania (CTC)	279
4.22.2	Modulacja szerokości impulsu (PWM)	285
4.22.3	Taktowanie licznika źródłem zewnętrznym	294
4.23	Magistrale komunikacyjne	295
4.23.1	SPI	297
4.23.2	TWI/TWSI (I ² C)	305
4.23.3	1-Wire	314
4.23.4	USART	321
4.24	Obsługa pamięci EEPROM	341
4.25	Przetwornik ADC	346
4.26	Moduły nadzorujące	358
4.27	Pisanie programów i poszukiwanie błędów	364

5	Przykłady zastosowań	376
5.1	Sterowanie diodami LED	376
5.2	Sterowanie brzęczykiem	380
5.3	Czujnik ruchu PIR	382
5.4	Silnik krokowy	386
5.5	Serwomechanizmy	392
5.6	Klawiatury	396
5.7	Wyświetlacz 7-segmentowy	400
5.8	Sterownik MAX7219 (matryce LED i wyświetlacze segmentowe) .	412
5.9	Wyświetlacz LCD (HD44780)	419
5.10	Czujnik temperatury DS18B20	442
5.11	Czujnik temperatury i wilgotności DHT11	451
5.12	Termistor jako czujnik temperatury	455
5.13	Czujniki gazów	461
5.14	Zegary czasu rzeczywistego DS1307, DS3231, PCF8563	473
A	Dodatek	479
A.1	Przedrostki	479
A.2	Konwersja jednostek	480
A.3	Systemy liczbowe	481
A.4	AVRDUDE (parametr -p)	482
A.5	Wyznaczanie współczynników charakterystyki czujników	484
A.6	Słowniczek angielsko-polski	486
	Skorowidz	494

Słowo wstępne

Niniejsza książka powstała na podstawie naszego przeszło 20-letniego doświadczenia w elektronice oraz programowaniu w kilkunastu językach. Na co dzień oboje pracujemy na Wydziale Fizyki Politechniki Warszawskiej na stanowiskach naukowo-dydaktycznych. Daje nam to styczność nie tylko z bieżącymi pracami naukowymi, ale również ze studentami, dzięki czemu mamy możliwość zaobserwowania, co sprawia największe trudności i które zagadnienia należy omówić dokładniej. Staraliśmy się, aby książka ta napisana była w sposób możliwie najprostszy, najbardziej przejrzysty i zrozumiały, a jednocześnie dostarczała jak największej ilości przydatnych informacji. Czy nam się to udało, ocenicie sami. Od kilku lat prowadzimy również kanał **Sala Science** na platformie YouTube, gdzie serdecznie Was zapraszamy. Znajdziecie tam wiele ciekawych filmów dotyczących również elektroniki i programowania. Spójrzcie na świat razem z nami z naukowego punktu widzenia.

W dzisiejszych czasach trudno już znaleźć urządzenia, w których układy oparte są jedynie na elektronice analogowej. W większości przypadków stosuje się programowalne układy cyfrowe. Nie zawsze jest to lepsze rozwiązanie z punktu widzenia parametrów urządzenia, jednak na pewno dużo prostsze i bardziej uniwersalne. Łatwiej i szybciej napiszemy program, niż zaprojektujemy skomplikowany układ oparty na bramkach i tranzystorach. Obecnie umiejętność korzystania z układów programowalnych jest podstawą projektowania zaawansowanych urządzeń elektronicznych. Jednymi z takich układów programowalnych są mikrokontrolery z rodziny AVR. Architektura ta daje szerokie możliwości, a przy tym jest realnie prosta, dzięki czemu idealnie nadaje się do nauki programowania. Musicie jednak pamiętać, że praca z mikrokontrolerami nie sprowadza się jedynie do programowania. Nasz układ wymaga odpowiedniego zasilania, podłączenia programatora, a w większości przypadków również podłączenia takich dodatkowych zewnętrznych układów jak czujniki temperatury, gazów, mierniki ciśnienia, wilgotności, radiodiody czy wyświetlacze. Dlatego też w niniejszej książce postanowiliśmy omówić zarówno programowanie w języku C, jak i podstawy elektroniki potrzebne do pracy z mikrokontrolerami. Nie jest to oczywiście pełny wykład z elektroniki ze wszystkimi szczegółami ani też podręcznik programowania w języku C. To przegląd najważniejszych informacji z dziedziny elektroniki i programowania,

które umożliwią Wam samodzielne wykonanie układów elektronicznych z wykorzystaniem mikrokontrolerów AVR - od początku do końca. Do zrozumienia zawartości tej książki nie jest wymagana jakakolwiek specjalistyczna wiedza. Jeżeli zdajecie sobie sprawę z tego, że do zaświecenia żaróweczki należy podłączyć dwa bieguny z baterii i zamknąć obwód, oraz wiecie, co to jest zmienna w języku C i jak ją zadeklarować, to już wystarczy. Reszta potrzebnych informacji znajduje się w książce. We wprowadzeniu omawiamy podstawowe zagadnienia, czym są mikrokontrolery AVR i jak one działają. Mówimy też o bezpieczeństwie, ponieważ jest ono najważniejsze, jak również odpowiadamy na często zadawane pytania. Rozdział drugi to krótkie wprowadzenie do elektroniki. Opisujemy tam podstawowe elementy i ich działanie, mówimy, czym jest prawo Ohma i jak go używać. Pokazujemy między innymi, jak lutować oraz jak korzystać z podstawowych przyrządów pomiarowych, takich jak woltomierze czy amperomierze. W rozdziale trzecim natomiast przedstawiamy najważniejsze zagadnienia elektroniczne w odniesieniu do mikrokontrolerów AVR, czyli między innymi to, jak podłączyć zasilanie, jakie są sposoby taktowania układów AVR i jak podłączyć programator. Rozdziały te, tak jak już wspomnieliśmy, nie są pełnym kursem elektroniki. To garść istotnych informacji potrzebnych do pracy z mikrokontrolerami. Pamiętajcie, że im lepiej poznacie podstawy, tym łatwiej będzie Wam tworzyć skomplikowane projekty. Tak jest ze wszystkim. Nauczenie się podstaw jest kluczem do zaawansowanych dokonań! Jeżeli chcielibyście rozszerzyć swoją wiedzę elektroniczną jeszcze bardziej, to zachęcamy Was do zajrzenia do takich książek jak *Sztuka elektroniki* Paula Horowitza i Winfielda Hilla [1] czy *Elektronika* Johna Watsona [2]. W książkach Piotra Góreckiego natomiast możecie znaleźć dużo informacji chociażby na temat wzmacniaczy operacyjnych [3] czy układów cyfrowych [4]. Kolejną pozycją, którą bardzo dobrze się czyta i która porusza wiele ciekawych zagadnień, jest książka Roberta Pease'a *Projektowanie układów analogowych* [5]. O samych mikrokontrolerach AVR traktują natomiast między innymi dwie doskonałe książki Tomasza Francuza *Język C dla mikrokontrolerów AVR* [6] oraz *AVR. Układy peryferyjne* [7]. Natomiast *Mikrokontrolery dla początkujących* Piotra Góreckiego [8] to już omówienie programowania mikrokontrolerów AVR w języku BASCOM, który jest oparty na języku BASIC. W rozdziale czwartym omawiamy programowanie. No wreszcie! Dowiedziecie się między innymi, z jakich środowisk programistycznych możecie skorzystać, jak zaprogramować układ,

czym są fusebity, typy danych oraz najważniejsze elementy mikrokontrolera, czyli rejestry. Dalej znajdziecie opis poszczególnych modułów mikrokontrolerów AVR. Rozdział piąty natomiast to przykłady, w których wykorzystujemy całą zdobytą wiedzę do tego, aby skonstruować proste układy elektroniczne. Zobaczycie, jak to wszystko działa w praktyce i jak wiele rzeczy można zrobić przy użyciu mikrokontrolerów w banalny sposób.

Chcielibyśmy się także tutaj wytłumaczyć z używania wielu nazw i skrótów anglojęzycznych. Jesteśmy przeciwnikami niepotrzebnego zapożyczania nazw i zaśmiania języka słowami, które mają polskie powszechnie stosowane odpowiedniki. Jednak w wielu przypadkach trudne jest znalezienie odpowiedniego słowa, a nawet jeżeli ono istnieje, to jest na tyle rzadko używane w danym kontekście, że staje się niezrozumiałe. To mniej więcej tak jak z dawną nazwą atomów. Proponowano i przez pewien czas nawet używano w Polsce słowa „niedzialek”, jako że atom jest niepodzielny. Nazwa nijak się nie przyjęła i może całe szczęście, bo mielibyśmy teraz fizykę niedziałkową i bomby niedziałkowe zamiast atomowych. Natomiast na myszkę komputerową mielibyśmy mówić manipulator stołokulotoczny¹. Takie były początki polskiej informatyzacji. Poza tym, co jest może jeszcze ważniejsze, większość dokumentacji i not katalogowych, z jakimi się zetkniecie, będzie właśnie w języku angielskim, z rzadka w języku rosyjskim czy polskim (głównie w przypadku starszych układów elektronicznych). Znajomość obcych słów pozwoli Wam lepiej zrozumieć dokumentację i poszerzy Wasze umiejętności tworzenia układów elektronicznych. Stosowanie przez nas nazw anglojęzycznych jest zatem naszym świadomym i celowym wyborem. Aby jeszcze bardziej ułatwić Wam korzystanie z obcojęzycznej dokumentacji, na końcu książki znajduje się krótki słowniczek angielsko-polski. W kodach źródłowych używamy również anglojęzycznych nazw zmiennych czy funkcji. Pisanie polskich nazw, szczególnie bez ogonków, bywa niezrozumiałe.

Książkę tę można czytać na kilka sposobów. Jeżeli dopiero zaczynacie przygodę z elektroniką i programowaniem układów scalonych, to zachęamy Was, aby wybrać najbardziej oczywisty sposób, czyli przeczytać ją od początku do końca. Powiało banałem. Jeżeli jednak uważacie, że dobrze znacie elektronikę, a chcielibyście nauczyć się programowania mikrokontrolerów, to zaczniście od rozdziału 3 opisującego jedynie elektronikę w kontekście mikrokontrolerów

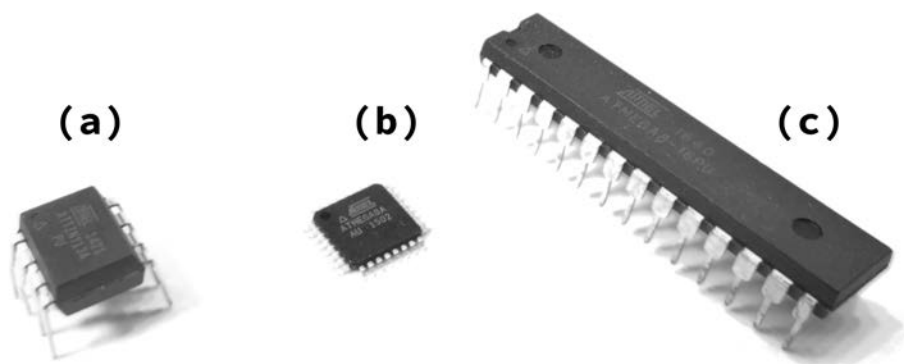
¹ Dawniej nie było myszek optycznych czy laserowych, ale zawierały one obracającą się kulkę.

AVR, a następnie przejdźcie do rozdziału 4 mówiącego o programowaniu. Jeżeli natomiast dobrze znacie elektronikę i programowanie w języku C, to możecie też zacząć od przykładowych zastosowań opisanych w rozdziale 5. W całej książce staraliśmy się, gdzie to tylko możliwe, umieszczać odpowiednie odnośniki do wcześniejszych rozdziałów, a w przypadku niektórych informacji powtarzać je wszędzie tam, gdzie są one pomocne. Dzięki temu nie będziecie musieli co chwilkę kartkować książki w przód i w tył. Miłej lektury!

1 Wprowadzenie

1.1 Czym są mikrokontrolery AVR?

Mikrokontrolery to zazwyczaj bardzo niewielkie komputery o zredukowanym zestawie instrukcji, czyli tzw. RISC (ang. *Reduced Instruction Set Computer*). Są to układy scalone (rysunek 1), wyposażone w procesor, wewnętrzną pamięć fizyczną RAM, pewien obszar pamięci flash oraz wiele innych modułów. Nie dziwi więc, że mikrokontrolery nazywamy czasami mikrokomputerami. Posiadają piny wejścia-wyjścia I/O (ang. *Input/Output*), liczniki, nierzadko zegary czasu rzeczywistego, przetworniki analogowo-cyfrowe czy komparatory. Jednocześnie są to zazwyczaj układy o bardzo niewielkich rozmiarach oraz niewygórowanych cenach. Doskonale sprawdzają się zatem przy tworzeniu różnego rodzaju układów, sterowaniu większymi urządzeniami oraz monitorowaniu parametrów, takich jak temperatura, wilgotność, ciśnienie czy prędkość obrotowa. Ich zastosowania są niemal nieograniczone. Niska cena, powszechna obecnie dostępność oraz łatwość w programowaniu czynią je bardzo istotnymi elementami dzisiejszej elektroniki nie tylko w zastosowaniach komercyjnych, ale również amatorskich.



Rysunek 1. Mikrokontrolery z rodziny AVR: (a) ATtiny13, (b) ATmega8 w obudowie do montażu powierzchniowego SMT (ang. *Surface-Mount Technology*), (c) ATmega8 w obudowie DIP27 do montażu przewlekane go THT (ang. *Through-Hole Technology*)

1 WPROWADZENIE

Od czasu pierwszych mikrokontrolerów, które pojawiły się na początku lat 70. XX wieku, powstało bardzo wiele rodzin mikrokontrolerów, a jedną z nich jest AVR. Ideę oraz pierwsze układy zaprojektowali dwaj studenci Norweskiego Instytutu Technologii NTH (nor. *Norges Tekniske Høgskole*), Alf-Egil Bogen oraz Vegard Wollan. Pierwszy mikrokontroler AVR został wyprodukowany przez firmę Atmel i trafił na rynek w 1997 roku, a był to AT90S1200. Nazwa AVR według niektórych pochodzi od imion twórców **Alf-Vegard-RISC**, jednak firma Atmel temu zaprzecza i podtrzymuje, że jest to nazwa rodziny mikrokontrolerów, a nie akronim. W 2016 roku produkcję mikrokontrolerów AVR przejęła firma Microchip Technology, która jest jednocześnie producentem mikrokontrolerów z rodziny PIC. Chociaż istnieją 32-bitowe mikrokontrolery AVR (AVR32), to wciąż najbardziej popularne, szczególnie w zastosowaniach amatorskich, pozostają mikrokontrolery 8-bitowe. Tymi właśnie mikrokontrolerami zajmiemy się w niniejszej książce.

1.2 Jak to wszystko działa?

No właśnie, ale jak to wszystko właściwie ma działać? W nie tak odległych czasach, gdy chcieliśmy sterować jakimś urządzeniem przy użyciu elektroniki, stosowało się bramki logiczne oraz przeróżnej maści układy scalone realizujące najrozmaitsze funkcje – od zliczania wartości, przez dzielenie częstotliwości, po zamianę kodu dziesiętnego na kod zrozumiały dla wyświetlaczy segmentowych. Wad tych rozwiązań było kilka. Zaczniemy od tego, że najprostsze nawet zadanie wymagało zastosowania bardzo wielu układów. Z jednej strony pociągało to za sobą koszty ich zakupu, a z drugiej wymagało niemałych umiejętności i włożenia dużej ilości pracy w zaprojektowanie i poskładanie całego urządzenia. Nie mówiąc już o tym, że wszystko to zajmowało bardzo wiele miejsca i pobierało niemałe ilości prądu, przez co o zasilaniu bateryjnym w wielu przypadkach można było tylko pomarzyć. Ponadto, gdy zaprojektowaliśmy już układ do sterowania jakimś urządzeniem, to jego przeprojektowanie czy wzbogacenie o nowe funkcje wymagało również wiele pracy. Tutaj z pomocą przychodzą mikrokontrolery. Mają małe rozmiary, są relatywnie tanie w stosunku do swoich możliwości, a to, jakie zadanie spełniają, zależy tylko od tego, jak napisaliśmy program. Dzięki nim tworzenie nawet skomplikowanych projektów stało się dużo łatwiejsze. Do działania mikrokontrolera wystarczy bowiem podłączyć

jedynie zasilanie oraz kilka kondensatorów filtrujących napięcie i gotowe. Aby mikrokontroler coś robił – poza tym, że ładnie wygląda – należy wgrać do niego odpowiedni program, czyli opis, co ma robić. W tym celu należy podłączyć do mikrokontrolera programator, czyli specjalne urządzenie, które umożliwi nam wysyłanie danych z poziomu komputera. Kod programu piszemy na komputerze, następnie kompilujemy, a skompilowany program wgrywamy do mikrokontrolera za pomocą programatora. To tak w telegraficznym skrócie. W ogólności nie mamy możliwości uruchomienia skompilowanego dla mikrokontrolera programu na naszym komputerze. W niektórych środowiskach jednak dostępne są specjalne symulatory, które – jak sama nazwa wskazuje – symulują działanie mikrokontrolera. Możemy wtedy, nawet nie posiadając mikrokontrolera, zobaczyć, jak będzie działał nasz program, co będzie się działo na pinach mikrokontrolera itd. Należy mieć jednak świadomość, że nigdy żaden symulator czy inne narzędzie służące do przeprowadzania symulacji rzeczywistego układu nie będzie w stanie zasymulować jego działania w stu procentach. Nie znaczy to, że symulatory nie są pomocne, oczywiście, że są, ale należy znać ich ograniczenia.

Wróćmy do naszego mikrokontrolera. Jesteśmy na etapie, w którym umieściliśmy nasz program w pamięci mikrokontrolera. Konkretnie wylądował on w nieulotnej pamięci flash. Nieulotna oznacza tyle, że po wyłączeniu zasilania czy pojawieniu się sygnału reset program nadal będzie się znajdował w pamięci i nie zostanie z niej usunięty, a co za tym idzie – nie musimy za każdym razem wgrywać programu na nowo. Pamięć flash jest odpowiednikiem takiego niewielkiego dysku twardego w komputerze. Pomimo że wyłączamy komputer, to nadal na dysku znajdują się nasze dane oraz system operacyjny. W przypadku AVR-ów akurat systemu operacyjnego nie ma. Procesor znajdujący się w mikrokontrolerze wykonuje jedynie nasz program instrukcja po instrukcji. Mamy zatem pełną kontrolę nad tym, co się dzieje w naszym mikrokontrolerze. W przypadku układów posiadających system operacyjny może się pojawić wiele problemów, których tutaj na szczęście nie uświadczymy. Skoro nasz program jest już wgrany do mikrokontrolera, to zostanie on uruchomiony za każdym razem, gdy włączymy jego zasilanie. Nie ma tutaj systemu operacyjnego ani mechanicznego dysku twardego, a zatem mikrokontroler możemy włączać i wyłączać w dowolnym momencie. Jedynie, jeżeli podłączamy do mikrokontrolera jakieś dodatkowe układy czy całe moduły, to powinniśmy najpierw wyłączyć zasilanie. Głównie po to, aby przypadkowe zwarcie jakichś wyprowadzeń nie

1 WPROWADZENIE

doprowadziło do uszkodzenia. Teraz nasuwa się Wam na pewno jeszcze bardzo wiele pytań. Na część z nich możecie znaleźć odpowiedź w rozdziale 1.4, ale najpierw przeczytajcie rozdział na temat bezpieczeństwa. Nie jest on długi, a może uchronić Was i Waszych bliskich czy znajomych przed wieloma niebezpieczeństwami.

1.3 Bezpieczeństwo

Zabawa z elektroniką niesie ze sobą nie tylko przyjemności, ale także pewne zagrożenia. Świadomość ich istnienia oraz stosowanie odpowiednich zasad bezpieczeństwa są konieczne, aby cieszyć się tą niezwykle ciekawą dziedziną. Nie pomijajcie tego rozdziału, tylko przeczytajcie go bardzo uważnie. Podajemy tutaj kilka najistotniejszych zasad, których należy bezwzględnie przestrzegać podczas pracy z urządzeniami oraz elementami elektronicznymi.

1. Wszelkie urządzenia elektryczne traktujcie jako potencjalne zagrożenie porażaniem prądem. Zwracajcie w szczególności uwagę na niezaizolowane przewody, iskrzące elementy i uszkodzone obudowy. Chociaż wielu osobom wydaje się, że urządzenia elektryczne i elektroniczne w dzisiejszych czasach są w pełni bezpieczne, wciąż dochodzi do bardzo wielu śmiertelnych wypadków spowodowanych porażeniem prądem. Jeżeli nie macie doświadczenia, nie zaczynajcie od projektowania układów pracujących pod napięciem sieciowym 230 V. Rozpocznijcie od układów o napięciu bezpiecznym, powiedzmy, poniżej 30 V prądu stałego. Z prądem należy uważać, w historii odnotowano przypadki śmiertelne nawet przy tak niskim napięciu jak 42 V prądu stałego [9]. W Polsce w latach 2005–2009 na skutek porażenia prądem zginęły 654 osoby [10], natomiast w Stanach Zjednoczonych rocznie ginie z tego powodu około tysiąca osób. Problem jest bardzo poważny, jednak przez wielu lekceważony.
2. Skóra ludzka ma dużo wyższą rezystancję niż organy wewnętrzne. Stąd też różnego rodzaju uszkodzenia skóry czy skaleczenia stanowią poważne zagrożenie. Należy je dobrze opatrzyć, chociażby przy użyciu plastrów.
3. Gdy pracujecie z urządzeniami czy układami elektronicznymi, Wasze ręce powinny być suche. Wilgoć powoduje zmniejszenie się rezystancji, a co za

tym idzie – zwiększa prawdopodobieństwo i pogarsza skutki porażenia prądem.

4. Wszelkie prace wykonujcie przy wyłączonym zasilaniu, a w przypadku urządzeń zasilanych z sieci przy wyjętym z gniazda przewodzie zasilającym. Ponadto pamiętajcie, że w niektórych urządzeniach wysokie napięcie może być wciąż obecne po odłączeniu zasilania, a nawet po wyciągnięciu przewodu z gniazda zasilania. Kondensatory wysokonapięciowe wciąż mogą być naładowane i stanowić duże zagrożenie porażenia prądem. Szczególnie groźne są tutaj duże kondensatory wysokonapięciowe o znaczącej pojemności! Historia zna przypadki, w których naładowane wysokonapięciowe kondensatory stały się przyczyną śmierci.
5. Zadbajcie o to, aby instalacja elektryczna w Waszym domu była poprawnie wykonana oraz posiadała zarówno zabezpieczenia nadprądowe (czyli bezpieczniki, żargonowo zwane korkami), jak i zabezpieczenia różnicowo-prądowe (zwane żargonowo różnicówkami). Jakiegolwiek prace, naprawy czy zmiany instalacji elektrycznej należy zlecić osobie, wykwalifikowanej posiadającej odpowiednie uprawnienia nadawane przez SEP (Stowarzyszenie Elektryków Polskich założone w 1919 roku). Ponadto urządzenia, których przewody zasilające mają 3 wyprowadzenia wymagają podłączenia do uziemienia, a zatem powinny być podłączane wyłącznie do gniazda z „bolcem”, czyli z przewodem PE lub PEN. W przeciwnym razie mogą stwarzać zagrożenie porażenia prądem. Poprawnie wykonana instalacja elektryczna zmniejsza prawdopodobieństwo porażenia prądem oraz pożaru!
6. Nigdy nie przecinajcie, nie otwierajcie ani nie szlifujcie elementów elektronicznych, w szczególności tranzystorów czy układów scalonych. Zawarte w nich substancje są bardzo często silnie toksyczne. W środku nie ma niczego ciekawego, co warto byłoby zobaczyć, za to można się porządnie zatruć.
7. Nie tnijcie, nie szlifujcie ani nie uszkodzajcie również elementów ceramicznych. Niektóre rezystory bardzo wysokiej mocy, tranzystory, diody wysokoczęstotliwościowe, podłoża układów, podkładki termoprzewodzące o bardzo dużej przewodności cieplnej, izolatory magnetronów stoso-

1 WPROWADZENIE

wanych w mikrofalówkach, a także niektóre obudowy żarówek LED (na przykład XLEDIA) są wykonywane na bazie tlenku berylu BeO (ang. *beryllium oxide*). Substancja ta ma wysoką przewodność cieplną, a przy tym jest dobrym izolatorem elektrycznym. Nie jest to częste połączenie, ponieważ dobre izolatory elektryczne mają zazwyczaj niską przewodność cieplną. Niestety tlenek berylu jest silnie toksyczny, gdy dostanie się do płuc nawet w niewielkiej ilości prowadzi do ciężkiej choroby zwanej beryliozą, a w niektórych przypadkach nawet do śmierci. Choroby tej nie należy mylić z boreliozą, która jest odkleszczową chorobą wywoływaną przez bakterie. Tlenek berylu najczęściej występuje w postaci białej ceramiki. Gdy ceramika pozostaje w jednym kawałku, jest relatywnie bezpieczna, problem pojawia się, gdy zostanie uszkodzona, na przykład ułamana, dlatego obchodzenie się z takimi elementami wymaga skrajnej ostrożności. Ponieważ wizualne odróżnienie tlenku berylu od innej ceramiki jest wręcz niemożliwe, amerykańskie stowarzyszenie krótkofalowców ARRL (ang. *American Radio Relay League*) zaleca, aby każdą ceramikę traktować jako potencjalnie wykonaną z tlenku berylu [11]. Ponieważ substancja ta posiada dobre właściwości izolujące, a jednocześnie ma wysoką przewodność cieplną, jest używana także jako dodatek do niektórych past termoprzewodzących, które stosowane są przy mocowaniu radiatorów lub umieszczane są wewnątrz obudów elementów elektronicznych, na przykład tranzystorów. Alternatywą są dostępne na rynku pasty termoprzewodzące na bazie krzemu (Si), srebra (Ag), miedzi (Cu) lub tlenku cynku (ZnO), które są dużo bezpieczniejsze w użyciu. Tlenek berylu w różnej postaci można znaleźć między innymi w takich tranzystorach jak 2N3866, BLW34, BLW82, rosyjskich P3, P4 oraz innych tranzystorach germanowych.

8. Niektóre (znaczy nie wszystkie): sondy pomiarowe, śrubki i wyprowadzenia trymerów pojemnościowych, złącza wysokoczęstotliwościowe (na przykład SMA), gniazda podstawek precyzyjnych (DIL), droższe wtyki (na przykład bananowe) czy narzędzia beziskrowe mogą być wykonane z miedzi berylowej (BeCu) lub innego brązu berylowego (stopu zawierającego beryl), które są toksyczne, tak jak i większość związków berylu. Beryl w postaci metalicznej można również znaleźć w niektórych wysokiej klasy głośnikach, głównie wysokotonowych (ang. *tweeters*). Brąz berylowy –

zależnie od składu – ma kolor zbliżony do brązu (nieco jaśniejszy) lub miedzi (bardziej pomarańczowy). Gdy materiał jest w całości, nie stanowi większego zagrożenia, jednak jego uszkodzenie, wyginanie czy szlifowanie, a co za tym idzie – uwalnianie się cząsteczek materiału do powietrza stwarza poważne zagrożenie chorobami płuc, wspomnianą wcześniej berylizacją, a nawet śmiercią.

9. Jeżeli zdarzy Wam się, że jakiś element, układ scalony czy tranzystor przepali się i zacznie wydobywać się z niego dym, to nie panikujcie. Odłączcie jak najszybciej zasilanie, a następnie przewietrzcie pomieszczenie. Uszkodzony element natomiasz wyrzucicie. Każdemu, kto zajmuje się elektroniką, wcześniej czy później takie przygody się przytrafiają. Tutaj mała dygresja. Wszelkie elektrośmieci, w tym urządzenia elektroniczne, baterie, żarówki czy świetlówki, należy usuwać zgodnie z dyrektywą unijną WEEE (ang. *Waste of Electrical and Electronic Equipment*) z dnia 4 lipca 2012 roku oraz Ustawą z dnia 11 września 2015 roku o zużytym sprzęcie elektrycznym i elektronicznym. Oznacza to, że należy zanieść taki śmieć do odpowiedniego punktu odbioru. Czasami stosowne pojemniki znajdują się również w sklepach wielkopowierzchniowych. W przypadku niekompletnego sprzętu elektronicznego, czyli na przykład telewizora bez zasilacza czy podświetlenia, pralki bez silnika czy lodówki bez agregatu, jedynym rozwiązaniem jest tzw. PSZOK, czyli Punkt Selektywnej Zbiórki Odpadów Komunalnych. Tam przyjmują wszystko. Pozostałe punkty, w sklepach czy rozsiane tu i ówdzie punkty odbioru elektrośmieci, nie mają prawa przyjmować urządzeń niekompletnych. Po prostu odmówią Wam odebrania takiego sprzętu, tłumacząc się zakazami wynikającymi z ustawy i, co więcej, będą mieli rację. Odpowiedzialne usuwanie elektrośmieci jest bardzo ważne, ponieważ elektronika często zawiera w sobie bardzo wiele substancji toksycznych. Wiele układów scalonych opartych jest na bazie arsenu, w niektórych sprzętach (na przykład świetlówkach) stosowana jest rtęć, a wciąż w wielu miejscach można znaleźć ołów czy inne substancje szkodliwe. Źle zutylizowane, leżące na wysypiskach substancje przenikają do gleby, uwalniają się do środowiska, a potem my wszyscy albo tym oddychamy, albo zjadamy w jedzeniu. Usuwanie na przykład metali ciężkich ze środowiska jest bardzo długotrwałym procesem. Widać

1 WPROWADZENIE

to chociażby po zwiększonej zawartości rtęci w mięsie wielu ryb. To nie jest dobre rozwiązanie. Dlatego też starajcie się poprawnie usuwać elektrośmieci i to nie dlatego, że jakieś przepisy Was do tego zmuszają, ale dlatego, że jest to działanie odpowiedzialne. Od tego, jak się zachowacie, będzie w przyszłości zależało zdrowie i życie Wasze, Waszych najbliższych, a także potomnych pokoleń.

10. Podczas lutowania i wszelkich prac noście okulary ochronne. Rozgrzane do temperatury 350°C spoiwo lutownicze czy lecąca z dużą prędkością ucięta metalowa końcówka elementu nie są niczym przyjemnym dla ludzkiego oka. Okulary ochronne kosztują parę złotych, tak niewiele trzeba, aby uniknąć uszkodzenia lub nawet utraty wzroku.
11. Lutowanie elementów powinno być wykonywane zawsze w dobrze wentylowanym pomieszczeniu. Korzystne jest zastosowanie specjalnych wyciągów lub filtrów powietrza. Absolutnym minimum natomiast powinno być zastosowanie chociażby niewielkiego wentylatora (wiatraczka), który będzie rozwiewał opary, najlepiej w połączeniu z otwartym lub uchylonym oknem.
12. Gdy wykorzystujecie kondensatory elektrolityczne, sprawdźcie dokładnie ich polaryzację, czy na pewno plus podłączony jest do wyższego napięcia niż minus. Wstawienie takiego kondensatora odwrotnie spowoduje jego wybuch i wydzielenie się toksycznych oparów. Dlatego też podczas uruchamiania nowego urządzenia nigdy nie nachylajcie się nad układem. Szczególnie niebezpieczne są kondensatory tantalowe.
13. Przed włączeniem zasilania wykonanego układu elektronicznego sprawdźcie, czy napięcia znamionowe, czyli maksymalne dopuszczalne napięcia zasilania zastosowanych w układzie elementów, są wyższe niż napięcie panujące w obwodzie. W przeciwnym razie elementy ulegną uszkodzeniu, a w skrajnym przypadku dojdzie do wybuchu lub pożaru.
14. Nie otwierajcie baterii, szczególnie tych wykonanych na bazie litu (litowo-jonowych, litowo-polimerowych LiPoly itp.). Zawarty w nich lit w połączeniu z powietrzem gwałtownie reaguje, wydzielając duże ilości energii. To natomiast prowadzi do powstania szkodliwych oparów, a w licznych przypadkach również do pożaru.

15. Wszelkie rozpuszczalniki, w tym preparaty do czyszczenia, jak chociażby alkohol izopropylowy, są zazwyczaj silnie łatwopalne. Należy przechowywać je w przewiewnym i chłodnym miejscu, z dala od źródeł ciepła czy światła słonecznego.
16. Niektóre czujniki przechyłu wykonane są na bazie rtęci. Kulka rtęci zwiera lub rozwiera styki, zależnie od orientacji elementu. Również niektóre kontaktrony posiadają kontakty zwilżone rtęcią (ang. *mercury wetted*). Rtęć obecna jest także w różnego rodzaju świetłówkach. Należy obchodzić się z tymi elementami ostrożnie, tak aby nie uległy uszkodzeniu i rtęć nie wydostała się na zewnątrz. Zarówno bezpośredni kontakt, jak i opary są bardzo szkodliwe.
17. Nigdy nie patrzcie w wiązkę laserową ani nie kierujcie jej w stronę ludzi lub zwierząt. Nawet z pozoru słabe wskaźniki laserowe mogą stanowić poważne zagrożenie dla wzroku, powodując jego uszkodzenie, a nawet całkowitą ślepotę! Niebezpieczne mogą być również diody LED wysokiej mocy, szczególnie te pracujące w paśmie niewidzialnym dla oka ludzkiego, czyli w ultrafiolecie (UV) lub w podczerwieni (IR). Natomiast kompletną już głupotą jest celowe świecenie laserem komuś w oczy czy też w kierunku przejeżdżających samochodów, lecących samolotów lub helikopterów. To prosty przepis na poważne problemy, również z prawem!
18. Podczas pracy nad elektroniką nie spożywajcie jedzenia ani nie pijcie napojów, w szczególności alkoholu. Jeżeli czujecie się zmęczeni lub głodni, zróbcie sobie przerwę, zjedzcie coś, napijcie się dobrej herbaty. Tak będzie lepiej.
19. Miejsce pracy powinno być dobrze oświetlone. W Polsce normy [12] zalecają, aby oświetlenie przy pracy przy komputerze, jak również przy obróbce mechanicznej oraz montażu dokładnym miało natężenie nie mniejsze niż 500 lx (luksów), natomiast w przypadku kreślenia, montażu precyzyjnego czy malowania było nie mniejsze niż 750 lx. Współczynnik odwzorowania barw (R_a , inaczej CRI) powinien być również nie mniejszy niż 80.
20. W pobliżu miejsca pracy powinna się również znajdować łatwo dostępna gaśnica. Nie musicie sobie od razu wstawiać 20-kilogramowej gaśnicy CO₂ zajmującej pół pokoju. Wystarczy niewielka gaśnica proszkowa,

1 WPROWADZENIE

choćby samochodowa. Dobrym rozwiązaniem są również spraye gaśnicze sprzedawane w puszkach o niewielkiej pojemności, na przykład 500 ml. Spray czy niewielka gaśnica wystarczają do stłumienia pożaru w zarodku. Przy takich objętościach środka gaśniczego oczywiście nie uda Wam się ugasić rozwiniętego już pożaru. W Polsce w ostatnich latach co roku dochodzi do około 30 tys. pożarów w budynkach mieszkalnych zarówno jedno-, jak i wielorodzinnych.

21. Nigdy nie pracujcie, gdy jesteście bardzo zmęczeni. W takim stanie człowiek potrafi narobić wiele głupot. W efekcie możecie nie tylko uszkodzić siebie i innych, ale również zrujnować projektowany przez Was od dłuższego czasu układ. Zamiast cieszyć się z postępów w pracy będziecie musieli naprawiać to, co napsuliście, a w niektórych przypadkach nawet zacząć wszystko od początku. A jak myślicie, skąd to wiemy? Tak, z własnego doświadczenia, a zatem nie popełniajcie naszych błędów.
22. Po pracy umyćcie zawsze dokładnie ręce.

1.4 Często zadawane pytania

W tym rozdziale przedstawiamy odpowiedzi na kilka ciekawych i często pojawiających się pytań.

Czy po wyłączeniu zasilania należy ponownie wgrać program do mikrokontrolera?

Nie. Program jest wgrywany do pamięci flash, która nie wymaga podtrzymywania napięciem. Oznacza to, że po wyłączeniu zasilania program nadal jest w pamięci i zostanie uruchomiony przy kolejnym włączeniu mikrokontrolera.

Czy programator musi być cały czas podłączony do mikrokontrolera?

Zazwyczaj nie. Programator służy jedynie do wgrania programu do mikrokontrolera. Wyjątek stanowi przypadek, gdy korzystamy z programatora jako źródła zasilania, na przykład zasilamy układ poprzez programator podłączony do portu USB.

Jak wiele razy mogę wgrać program do mikrokontrolera?

Bardzo wiele razy. Zgodnie z dokumentacją mikrokontroler zazwyczaj powinien wytrzymać nie mniej niż 10 000 pełnych cykli, podczas których program jest usuwany i zapisywany.

Jak usunąć program z mikrokontrolera?

Zazwyczaj nie ma takiej konieczności. Aby usunąć program, można albo wgrać do mikrokontrolera nowy program, albo wyczyścić pamięć flash. Aby natomiast przerwać działanie programu, wystarczy wyłączyć zasilanie mikrokontrolera.

Jakim napięciem można zasilać mikrokontrolery AVR?

Większość mikrokontrolerów AVR może być zasilana napięciem nie większym niż 5 V (ewentualnie 5.5 V). Dolna granica jest różna, zależnie od modelu mikrokontrolera oraz jego wersji. Niektóre mikrokontrolery mogą być zasilane tak niskim napięciem jak 1.8 V, a inne wymagają na przykład zasilania od 4.5 V do 5.5 V.

Jak do mikrokontrolera zasilanego napięciem 5 V podłączyć moduł, którego napięcie zasilania wynosi 3.3 V?

Gdy napięcie zasilania mikrokontrolera jest różne od napięcia zasilania modułu, który chcemy wykorzystać, pojawia się problem. Istnieje obawa, że podczas komunikacji urządzenia nie będą rozumiały przesyłanych do siebie sygnałów. Problem ten możemy rozwiązać na kilka sposobów. Istnieją moduły, których maksymalne napięcie zasilania wynosi co prawda 3.3 V, jednak są kompatybilne z sygnałami TTL, czyli mogą odbierać sygnały o napięciu 5 V. W takim przypadku możemy piny sygnałowe modułu podłączyć bezpośrednio do mikrokontrolera. Sam moduł natomiast musi być oczywiście zasilany z niezależnego źródła o napięciu 3.3 V. To, czy nasz moduł ma taką możliwość, należy sprawdzić w nocie katalogowej lub zapytać o to sprzedawcę. Innym rozwiązaniem jest zastosowanie konwertera poziomów logicznych. Można go nabyć jako oddzielny moduł za parę złotych. Dzięki niemu po jednej stronie mamy poziomy logiczne na poziomie 0 V oraz 3.3 V, a po drugiej poziomy logiczne 0 V oraz 5 V. Konwerter taki można zazwyczaj wykorzystać również do konwersji na napięcia niższe niż 3.3 V. Podobnymi modułami są wykorzystywane do komunikacji USART (RS-232) konwertery, które zamieniają sygnały mikrokontrolera na sygnały o napięciach dostosowanych do magistrali RS-232. Trzecie rozwiązanie

1 WPROWADZENIE

jest takie, aby obniżyć napięcie zasilania mikrokontrolera do 3.3 V. Jest to niewątpliwie najprostsze rozwiązanie, jednak wtedy zastosowanie modułów o napięciu zasilania 5 V stanie się problematyczne.

Jaką wydajność prądową musi mieć źródło zasilania podłączone do mikrokontrolera (na przykład zasilacz)?

Do samego działania mikrokontroler wymaga prądów rzędu miliamperów, zatem dowolny zasilacz czy bateria spełnią to wymaganie. Przy doborze wydajności prądowej zasilacza należy się kierować wartością prądu maksymalnego układów podłączonych do mikrokontrolera. Sam mikrokontroler nie pobiera dużego prądu, ale podpięta do niego dioda LED, wyświetlacz LCD czy podłączony poprzez stopień mocy silnik prądu stałego już tak. Na dobry początek zasilacz powinien umożliwiać dostarczenie prądów rzędu 300 – 500 mA lub więcej.

Czy do pinu wejścia-wyjścia (I/O) mogę podłączyć bezpośrednio diodę lub silnik?

To zależy od maksymalnego prądu, jaki ten odbiornik pobiera. Jeżeli jest to niewielka dioda LED, która pobiera 20 mA prądu, to tak, można taką diodę podłączyć do pinu, jednak należy pamiętać o dodaniu w szeregu rezystora o wartości na przykład 1 k Ω , w celu ograniczenia maksymalnego prądu. Jeżeli jednak jest to dioda mocy lub silnik prądu stałego, które pobierają dużo wyższe prądy, rzędu kilkudziesięciu czy nawet kilkuset miliamperów, to takiego odbiornika nie wolno podpinąć do pinu wejścia-wyjścia, ponieważ może to spowodować uszkodzenie mikrokontrolera. Maksymalny prąd, jaki może zostać dostarczony przez pojedynczy pin, to zazwyczaj około 40 mA, jednak całkowity pobór prądu z mikrokontrolera nie może przekroczyć 200 mA.

Czy można przetaktować mikrokontroler?

Tak, mikrokontroler można nieznacznie przetaktować, czyli podłączyć do niego sygnał taktujący o częstotliwości wyższej niż opisana przez producenta. W takim przypadku jednak układ zacznie się przegrzewać, a jego działanie może być nieprzewidywalne. Dlatego też nie powinno się przekraczać częstotliwości taktowania podanej w nocie katalogowej. Istnieją oczywiście śmiałkowie, którym udało się na krótki czas uruchomić mikrokontroler przy znacznie zwiększonej częstotliwości, nawet do przeszło 65 MHz [13]. Dla mikrokontrolera pra-

cującego nominalnie z częstotliwościami do 20 MHz oznacza to przeszło 3 razy większą częstotliwość taktowania. Wymaga to jednak dołączenia odpowiedniego chłodzenia z wykorzystaniem na przykład ciekłego azotu. Podsumowując, możliwe jest przetaktowanie mikrokontrolera, jednak nie powinniście tego robić w żadnych swoich projektach, a tym bardziej w seryjnych produkcjach. Jest to co najwyżej ciekawostka. Wszystkim tym, którzy są jednak rządni wrażeń i dreszczyku emocji związanych z przetaktowywaniem, polecamy mimo wszystko nie przekraczać częstotliwości o więcej niż 5 % i mieć świadomość, że mikrokontroler może zacząć działać w sposób nieprzewidywalny, a nawet może dojść do jego uszkodzenia.

Czy na mikrokontrolerze AVR można wykonywać skomplikowane obliczenia?

Na mikrokontrolerach AVR możliwe jest wykonanie obliczeń zarówno stałoprzecinkowych, czyli na liczbach całkowitych, jak i zmiennoprzecinkowych, czyli na liczbach rzeczywistych (z przecinkiem). Należy mieć jednak na uwadze, że żaden mikrokontroler AVR nie posiada koprocatora zmiennoprzecinkowego, czyli specjalnego modułu, który ułatwia obliczenia zmiennoprzecinkowe. W efekcie obliczenia zmiennoprzecinkowe i tak wykonywane są przy użyciu liczb całkowitych, co powoduje, że są dość powolne. Ponadto mikrokontrolery mają niewielką ilość pamięci fizycznej oraz ograniczoną częstotliwość taktowania (maksymalnie 20 MHz), czyli mówiąc kolokwialnie, szybkość procesora. Oznacza to, że wykonanie skomplikowanych obliczeń lub operowanie na dużych zbiorach danych może być bardzo trudne, a na wynik obliczeń będziemy bardzo długo czekać. Z pewnością nie jest to architektura do wykonywania bardzo skomplikowanych symulacji czy obliczeń numerycznych. Z drugiej jednak strony wykonanie prostych obliczeń, rozwiązanie układu równań wielomianowych, wyznaczenie wartości funkcji czy nawet wykonanie szybkiej transformaty Fouriera (FFT) jak najbardziej leżą w zakresie możliwości mikrokontrolerów AVR.

PROGRAM PARTNERSKI

— GRUPY 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>

GRUPA
Helion

Interesujesz się elektroniką? Marzysz o konstruowaniu zaawansowanych urządzeń, w których migają wyświetlacze, furkoczą serwomechanizmy i silniki krokowe? Dopiero zaczynasz przygodę w tej dziedzinie, a może jesteś weteranem, który chce poszerzyć swoją wiedzę? Ta książka jest właśnie dla Ciebie! Dla kogoś, kto chce się rozwijać, kto zamierza od podstaw, gruntownie poznać i opanować jedną z najważniejszych umiejętności XXI wieku, jaką jest programowanie mikrokontrolerów. Dziś niemal wszystkie urządzenia bazują na układach programowalnych. Należą do nich 8-bitowe mikrokontrolery AVR, które w ostatnich latach zyskały sporą popularność. To o nich opowiadamy w tej książce.

Znajdziesz tu między innymi:

- Opis podstawowych elementów elektronicznych i ich oznaczeń
- Porady, jak czytać noty katalogowe i schematy elektroniczne, a także jak rozwiązywać problemy
- Wskazówki, jak poprawnie wykonywać pomiary multimetrem i oszacować błędy pomiarowe
- Instrukcje, jak lutować i rozlutowywać elementy, wykonać zasilanie, podłączyć programator i taktować mikrokontroler
- Charakterystykę środowisk programistycznych
- Omówienie modułów mikrokontrolera (rejstry, liczniki, magistrale komunikacyjne I2C, SPI, USART, przetwornik analogowo-cyfrowy ADC, obsługa pamięci EEPROM), a także magistrali 1-Wire
- Podstawy programowania mikrokontrolerów (przesunięcia bitowe, system heksadecymalny, typy danych, kod uzupełnień do dwójki itp.)
- Kody źródłowe w języku C dla mikrokontrolerów ATmega8/16/32, ATmega48/88/168/328, a czasami nawet dla ATtiny13
- Podpowiedzi, jak szukać błędów w programie (debugowanie)
- 14 gotowych zastosowań wraz z pełnymi kodami źródłowymi (wyświetlacze LCD, 7-segmentowe, silniki krokowe, serwomechanizmy, czujniki gazów, temperatury i wilgotności, zegary czasu rzeczywistego RTC)
- Techniczny słowniczek angielsko-polski

Zachęcamy również do odwiedzenia prowadzonego przez autorów kanału naukowego Sala Science na platformie YouTube!

dr inż. Filip Sala ukończył studia na Wydziale Fizyki Politechniki Warszawskiej, na kierunku fizyka techniczna ze specjalnością optoelektronika. W 2012 roku otrzymał stopień doktora nauk fizycznych, a od 2013 pracuje na stanowisku adiunkta naukowo-dydaktycznego na macierzystej uczelni. Autor wielu publikacji o zasięgu międzynarodowym z dziedziny optyki nieliniowej, analizy numerycznej i modelowania zjawisk fizycznych. Od przeszło 20 lat zajmuje się elektroniką — zarówno cyfrową, jak i analogową — a także programowaniem. W czasie wolnym fotografuje, jeździ rowerem i chodzi po górach.

dr inż. Marzena Sala-Tefelska ukończyła studia na Wydziale Fizyki Politechniki Warszawskiej, na kierunku fizyka techniczna ze specjalnością optoelektronika. W 2012 roku otrzymała stopień doktora nauk fizycznych i rozpoczęła pracę jako adiunkt naukowo-dydaktyczny na macierzystej uczelni. Autorka licznych publikacji międzynarodowych w dziedzinie telekomunikacji światłowodowej, optyki nieliniowej, ciekłych kryształów i niebieskiej fazy. Od wielu lat zajmuje się elektroniką i programowaniem. W czasie wolnym chodzi po górach, jeździ rowerem, fotografuje i maluje obrazy.

Helion 	Sprawdź nasze szkolenia!	KOD KORZYŚCI Sięgnij po więcej! ▶ 
 helion.pl	SZKOLENIA	
 HELION SA ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl	 AKADEMIA IT & BUSINESS WWW.SZKOLENIA.HELION.PL	ISBN 978-83-283-7625-0  9 788328 376250
INFORMATYKA W NAJLEPSZYM WYDANIU		Cena: 129,00 zł