

Spis treści

O autorze	xi
O recenzencie technicznym	xii
Podziękowania	xiii
Od autora	xv
1 Włączanie i wyłączenie diody LED za pomocą przycisku	1
Eksperyment	3
Sprzęt	3
Schemat układu: połączenie systemu DAQFactory i urządzenia LabJack	4
Oprogramowanie	5
Przebieg eksperymentu, testy i proces projektowania	13
Analiza	15
Wykorzystanie systemu Raspberry Pi i języka Python w celu sterowania świeceniem diody LED	16
Eksperyment	17
Przebieg eksperymentu	19
Analiza	20
Listingi kodów	20
Podsumowanie	22
2 Sterowanie zasilaniem, monitorowanie mocy i tworzenie wyspecjalizowanych interfejsów graficznych	23
Eksperyment	25
Sprzęt	28
Oprogramowanie	29

Wymagane kontrolki ekranowe	29
Przebieg eksperymentu, testowanie i projektowanie	31
Analiza	33
Monitorowanie prądu i zarządzanie zasilaniem za pomocą Raspberry Pi	43
Eksperyment	45
Przebieg eksperymentu	47
Analiza	48
Listingi kodów	48
Podsumowanie	64
3 Wprowadzenie do skryptów	65
Eksperyment	66
Sprzęt	66
Oprogramowanie	67
Skrypty	68
Przebieg eksperymentu	70
Analiza	70
Sekwencje DAQFactory – sterowanie zestawem diod LED	
podłączonych do Arduino	71
Eksperyment	71
Analiza	71
Raspberry Pi	72
Listingi kodów	73
Podsumowanie	78
4 Wprowadzanie danych	79
Sprzęt 80	
Oprogramowanie	81
Wymagane kontrolki ekranowe	81
Skrypty	86
Przebieg eksperymentu	87
Analiza	87
Wprowadzanie danych przy użyciu mikrokontrolera Arduino	88
Eksperyment	89

Przebieg eksperymentu	92
Analiza	92
Raspberry Pi – wprowadzanie danych z ekranu	93
Listingi kodów	94
Podsumowanie	109
5 Sygnały i wyjścia cyfrowe	111
Eksperyment	112
Sprzęt	112
Oprogramowanie	114
Przebieg eksperymentu	115
Analiza	115
Sterowanie za pomocą środowiska DAQFactory zestawem diod LED podłączonym do płytki mikrokontrolera	116
Eksperyment	116
Przebieg eksperymentu	117
Analiza	117
Raspberry Pi	118
Listingi kodów	119
Podsumowanie	127
6 Przetwarzanie analogowo-cyfrowe i cyfrowo-analogowe danych wejściowych i wyjściowych	129
Konwersja cyfrowo-analogowa	133
Eksperyment – konwersja z systemu dziesiętnego na binarny z wykorzystaniem urządzenia LabJack i systemu DAQFactory	134
Sprzęt	134
Oprogramowanie	134
Konwersja analogowo-cyfrowa	135
Przebieg eksperymentu	137
Analiza	139
Konwersja analogowo-cyfrowa z zastosowaniem mikrokontrolerów	142
Eksperyment	142
Przebieg eksperymentu	146

Analiza	147
Projektowanie systemu i programowanie	148
Konwersja wejściowych i wyjściowych sygnałów analogowych i cyfrowych przy użyciu urządzenia Raspberry Pi	148
Konwersja z systemu dziesiętnego na binarny	149
Konwersja analogowo-cyfrowa z wykorzystaniem Raspberry Pi	150
Eksperyment	150
Przebieg eksperymentu	151
Analiza	153
Listingi kodów	156
Listingi kodów dla Raspberry Pi	167
Podsumowanie	174
7 Zarządzanie poziomem mocy i intensywnością oświetlenia	175
Eksperyment	177
Sterowanie napięciem	177
Eksperyment	181
Sterowanie napięciem za pomocą metody PWM	181
Przebieg eksperymentu	182
Analiza	184
Sterowanie intensywnością oświetlenia za pomocą Raspberry Pi	184
Wprowadzenie	184
Eksperyment	186
Przebieg eksperymentu	189
Analiza	191
Listingi kodów	192
Podsumowanie	198
8 Zdarzenia i odmierzanie czasu	199
Określanie i odmierzanie czasu za pomocą oprogramowania	200
Podstawowe zmienne, używane do wyznaczania czasu	201
Uruchamianie zaplanowanych zdarzeń	202
Stoper	202

Sprzętowe odmierzenie czasu, zarządzanie zdarzeniami i wyznaczenie częstotliwości	204
Eksperyment	207
Sprzęt	207
Schemat układu	207
Oprogramowanie	207
Skrypty	209
Układ elektroniczny	209
Oprogramowanie	210
Przebieg eksperymentu	210
Analiza	211
Określanie i odmierzenie czasu za pomocą sprzętu	212
Eksperyment	214
Schemat	215
Przebieg eksperymentu	215
Analiza	216
Zegary mikrokontrolera, obsługa czasu i zliczanie zdarzeń	218
Eksperyment	219
Przebieg eksperymentu	220
Analiza	221
Zliczanie zdarzeń i obsługa czasu za pomocą Pythona i urządzenia Raspberry Pi	222
Planowanie zdarzeń	224
Wykrywanie i zliczanie zdarzeń	225
Eksperyment	228
Przebieg eksperymentu	231
Analiza	235
Listingi kodów	236
Programy Pythona, uruchamiane w urządzeniu Raspberry Pi	244
Podsumowanie	253
9 Graficzna prezentacja danych	255
Eksperyment – liniowe wykresy graficzne	257
Część 1. Wybór sprzętu i komponentów – prostokątny przebieg wyjściowy	257
Wymagane elementy elektroniczne	258

Schemat układu	258
Oprogramowanie	259
Wymagane kontrolki ekranowe	259
Przebieg eksperymentu	262
Część 2. Wybór sprzętu i komponentów – piłokształtne i trójkątne przebiegi wyjściowe	264
Przebieg eksperymentu	264
Część 3. Wybór sprzętu i komponentów – przebieg trójkątny o zboczu narastającym i opadającym	266
Przebieg eksperymentu	268
Tworzenie wykresów x-y	269
Przebieg eksperymentu	272
Analiza	274
Tworzenie wykresów czasowych	275
Wyświetlanie wykresów x-y	280
Wyświetlanie wykresów za pomocą mikrokontrolera	281
Eksperyment	282
Przebieg eksperymentu	283
Analiza	283
Graficzna prezentacja danych za pomocą Pythona i Raspberry Pi	284
Wprowadzenie	284
Eksperyment	285
Przebieg eksperymentu	287
Analiza	290
Listingi kodów	292
Podsumowanie	294
10 Sterowanie prądem	295
Źródła prądu stałego	296
Eksperyment	297
Sprzęt	297
Oprogramowanie	300
Przebieg eksperymentu	300
Analiza	301

Sterowanie większymi prądami stałymi	302
Silniki bezszczotkowe prądu stałego (bez komutatorów i iskrzących szczotek) . . .	302
Eksperyment	303
Sprzęt	303
Schemat obwodu	305
Oprogramowanie	307
Przebieg eksperymentu	307
Analiza	308
Silniki krokowe	310
Eksperyment	311
Sprzęt	311
Schemat układu	312
Oprogramowanie	314
Przebieg eksperymentu	315
Analiza	316
Sterowanie źródłami prądu przemiennego	318
Wprowadzenie	318
Eksperyment	320
Sprzęt	320
Schemat obwodu	322
Oprogramowanie	323
Przebieg eksperymentu	324
Analiza	324
Sterowanie prądem za pomocą Raspberry Pi i Pythona	327
Sterowanie większymi prądami stałymi	327
Eksperyment	329
Przebieg eksperymentu	331
Analiza	331
Sterowanie mocą w przypadku obciążeń indukcyjnych	333
Wprowadzenie	333
Eksperyment	333
Przebieg eksperymentu	338
Analiza	338
Sterowanie prądami przemiennymi	339

Wprowadzenie	339
Eksperyment	339
Przebieg eksperymentu	341
Analiza	342
Listingi kodów	343
Kody języka Python dla Raspberry Pi	343
Podsumowanie	345
11 Mikrokontrolery i komunikacja szeregową	347
Eksperyment – przesyłanie danych z mikrokontrolera do komputera głównego	351
Sprzęt	353
Oprogramowanie	354
Wymagane kontrolki ekranowe	358
Przebieg eksperymentu	358
Analiza	359
Eksperyment – przesyłanie danych z komputera głównego do mikrokontrolera	362
Wprowadzenie	362
Sprzęt	362
Oprogramowanie	362
Przebieg eksperymentu	364
Analiza	364
Raspberry Pi i Arduino	365
Eksperyment	366
Przebieg eksperymentu	367
Analiza	369
Listingi kodów	370
Podsumowanie	374
A Lista skrótów	375
B Lista producentów	379
Indeks	381

O autorze



Richard J. Smythe studiował chemię z elementami matematyki i fizyki na Uniwersytecie Brocka, położonym na południu prowincji Ontario. Studia ukończył z wyróżnieniem po czterech latach. Następnie zdobywał wiedzę na Uniwersytecie Waterloo, na którym uzyskał tytuł magistra chemii analitycznej i informatyki oraz doktorat z chemii analitycznej. Po odbyciu stażu podoktorskiego na Uniwersytecie Stanowym Nowego

Jorku w Buffalo w zakresie chemii elektroanalitycznej założył w 1974 roku firmę Peninsula Chemical Analysis Ltd., specjalizującą się w analizach chemicznych. Już w 1966 roku, zanim jeszcze nauczył się języka Fortran IV, na uniwersytecie Waterloo poznał metodę wykonywania obliczeń z podziałem czasu za pomocą taśm papierowych, kart perforowanych i języka BASIC. Interesował się zarówno minikomputerami PDP 11, jak i komputerami typu PC, a także zajmował się programowaniem urządzeń używając różnych języków komputerowych i skryptowych. Wcześniej był właścicielem komercyjnego laboratorium, a obecnie jest konsultantem specjalizującym się w chemii analitycznej, specjalistą kryminologiem w PCA Ltd., pełnoprawnym partnerem w spółce Walters Forensic Engineering w Toronto (Ontario), a także głównym naukowcem w Contrast Engineering Limited w Halifax (Nowa Szkocja). Duża część kariery zawodowej Richarda polega na opracowywaniu metod, dzięki którym problem ostatecznie wymaga wykonania jednego lub więcej pomiarów podstawowych i może zostać rozwiązany przy użyciu sprzętu dostępnego pod ręką lub zakupionego w zwykłym sklepie.

O recenzencie technicznym



Roland Meisel uzyskał tytuł licencjata z fizyki na Uniwersytecie Windsor, licencjat z pedagogiki ze specjalizacją w fizyce i matematyce na Uniwersytecie Queen oraz tytuł magistra inżyniera z fizyki na Uniwersytecie Waterloo. Zanim zajął się nauczaniem, pracował w Chalk River Nuclear Laboratories. Następnie przez dwadzieścia osiem lat uczył fizyki, matematyki i informatyki w szkołach średnich prowincji Ontario. Będąc kierownikiem wydziału matematyki w szkole Ridgeway

Crystal Beach High School przeszedł na emeryturę, na której zajął się twórczością wydawniczą. Wspierał proces publikowania książek poświęconych różnym zagadnieniom matematycznym i fizycznym, poczynając od wstępu do algebry, a kończąc na rachunku różniczkowym. Był między innymi recenzentem technicznym, autorem, twórcą interaktywnych plików internetowych (które zaprojektował, utworzył, opublikował i edytował) oraz fotografem. Udziela się w kilku organizacjach, w tym w Ontaryjskim Stowarzyszeniu Nauczycieli Fizyki, Ontaryjskim Stowarzyszeniu Nauczycieli Matematyki, Kanadyjskim Stowarzyszeniu Pilotów i Właścicieli Samolotów oraz Towarzystwie Historycznym Wainfleet.

Zawsze bardzo interesował się techniką, a swój pierwszy komputer osobisty, Apple II z procesorem 1 MHz i 16 kB pamięci, zamówił pocztą z Kalifornii w 1979 roku. W wolnym czasie można go spotkać pilotującego małe samoloty, jeżdżącego na rowerze lub motocyklu, obrabiającego drewno, czytającego czy też grającego na pianinie i innych instrumentach.

Podziękowania

Podziękowania chciałbym rozpocząć od wspomnienia o moich zmarłych rodzicach, Richardzie H. Smythe i Margaret M. Smythe (z domu Earle), którzy po wojnie wyemigrowali z Londynu zabierając ze sobą trójkę dzieci i ostatecznie wychowali ich czwórkę w Kanadzie. Nasi rodzice zaszczepili w nas potrzebę zdobycia jak najlepszego wykształcenia, aby każdy był samowystarczalny i niezależny. Ta niezależność pozwoliła dwom z nas cieszyć się z odpowiednio wysokiej emerytury, a najmłodszemu z rodzeństwa kontynuować pracę w wybranym zawodzie przez blisko dekadę po przejściu na emeryturę. Najstarszy jest cały czas aktywnym konsultantem, specjalizującym się w chemii analitycznej oraz zajmuje się cywilną kryminalistyką sądową.

Przez całe moje życie wiele osób było moimi mentorami, inspirowało mnie, uczyło, przekazywało wiedzę, sztukę racjonalnego myślenia, wytrwałości, a w większości przypadków cenną mądrość. Oto one:

- ◆ E. Glyn-Jones – matematyczka; J. A. Smith – dyrektor; E. Umbrico – fizyk (szkoła Merritton High School w St. Catharines, Ontario).
- ◆ Prof. E. A. Cherniak, prof. R. H. Hiatt, prof. F. Koffyberg oraz prof. J. M. Miller (Uniwersytet Brocka w St. Catharines, Ontario).
- ◆ Prof. G. Atkinson (Uniwersytet Waterloo w Waterloo, Ontario).
- ◆ Prof. S. Bruckenstein (Uniwersytet Stanowy Nowego Jorku w Buffalo).

Można by powiedzieć, że pierwszy etap zdobywania wiedzy rozpoczął się wtedy, gdy jako rodzic upewniłem się, iż obie moje córki, Wendy i Christie, potrafią już czytać w bardzo młodym wieku. Wówczas opracowałem dla nich graficzne pomoce naukowe, pozwalające na naukę i zrozumienie binarnej arytmetyki cyfrowej.

Podziękowania nie byłyby kompletne bez wzmiankowania osoby, która pomimo codziennego chaosu związanego z życiem na wsi sprawiła, że mogłem dysponować czasem potrzebnym do pisania. Mam na myśli moją małżonkę, Lindę. Przez wiele lat cierpliwie zносиła eksperymenty, a także papiery, notatki, książki, deski kreślarskie

skie, przewody, elementy elektroniczne porzucane wszędzie, gdzie się da (gdy nie patrzyła, nawet na stole kuchennym)! Dziękuję ci, moja droga.

Chociaż moje życie zawodowe polega na rozwiązywaniu problemów, związanych z dziedziną chemii i tworzeniu raportów wyjaśniających, jak powstały oraz w jaki sposób można naprawić ich skutki lub uniknąć ponownego wystąpienia, nigdy nie napisałem żadnej książki. Niniejsza pozycja nie mogłaby powstać bez pomocy i wskazówek redaktorów wydawnictwa Apress – pani Natalie Pao, pani Jessiki Vakili i pana Marka Powersa.

Od autora

Książka *Arduino w nauce* została napisana w celu zaprezentowania podstawowych technik, które mogą być wykorzystywane w trakcie nauki opartej na eksperymentach. Miejmy nadzieję, że dzięki tej pracy studenci i osoby, które dopiero zaczynają się zajmować elektrotechniką lub naukami fizycznymi (albo mają ograniczone doświadczenie związane z nimi), będą mogły projektować i przeprowadzać eksperymenty niezbędne do dalszego rozwoju i edukacji. Można też założyć, że książka ta będzie przydatna w przypadku dysponowania ograniczonymi środkami finansowymi przeznaczonymi na opracowywanie eksperymentów i programów eksperymentalnych lub edukacyjnych.

Przemieszczające się lub żerujące zwierzęta (w tym owady) wykorzystują światło dzienne, zbliżone do podczerwieni lub spolaryzowane, punkty na niebie, ślady chemiczne w wodzie, pole magnetyczne Ziemi i inne środki podczas poruszania się w poszukiwaniu pożywienia lub powrotu na swoje tereny lęgowe. Astronomia, biologia, chemia, geologia/geografia, matematyka, fizyka i inne przedmioty aż po zoologię są naukami utworzonymi przez ludzi, całkowicie nieznanymi podróżnikom ze świata zwierząt. Istnieją jednak pewne podobieństwa pomiędzy wykorzystaniem przez królestwo zwierząt wielu zjawisk, o których nie mają one pojęcia, a obecnymi badaniami naukowymi. Znacząca ilość pozyskiwanej wiedzy naukowej jest odkrywana przez badaczy specjalizujących się w jednej dyscyplinie, wykorzystujących nieznaną im techniki eksperymentalne z innej dziedziny. Ta książka, chociaż napisana przez chemika analitycznego, jest kompilacją podstawowych metod mających zastosowanie w każdej dyscyplinie naukowej, która wymaga eksperymentalnych pomiarów typowych parametrów fizykochemicznych.

Autor jest doświadczalnym chemikiem analitycznym, który zajmował się lampami próżniowymi, tranzystorami, układami scalonymi, komputerami mainframe, minikomputerami, mikrokomputerami i mikrokontrolerami, a także językami programowania BASIC, Fortran i różnymi odmianami C włącznie z produktami o otwartych źródłach, takimi jak Python i Processing (język zintegrowanego środowiska programistycznego IDE dla mikrokontrolera Arduino), jak również wersjami systemów operacyjnych Linux

używanymi w Raspberry Pi. Wersje tych języków, środowisk programistycznych i systemów operacyjnych są dostępne bezpłatnie w internecie i podlegają ciągłym zmianom.

Tę książkę można byłoby potraktować za faktycznie przestarzałą już w momencie jej pisania, ale podobnie jak nauka i technika, którą opisuje, jest ona punktem wyjścia w ciągle zmieniającym się świecie technologii. Podstawy nauki są jednak dość niezmiennie i dobrze zrozumiałe dla badaczy i naukowców-praktyków, więc należy zachować dużą ostrożność podczas twierdzenia, że jakaś koncepcja lub technika jest „przestarzała”. Systemy SCADA i ich rozwój znacznie wyprzedziły powstanie komputerów PC. Niektóre z układów scalonych, wyprodukowanych w technologii tranzystorów bipolarnych i CMOS (włącznie z serią 7400), które są dziś w powszechnym użyciu, pochodzą z lat 70. XX wieku. Wiele technik analizy chemicznej i pomiarów fizycznych, obecnie nauczanych i używanych, pochodzi w rzeczywistości ze średniowiecza.

SCADA to akronim (ang. *Supervisory Control And Data Acquisition*) oznaczający nadzorowanie procesu technologicznego i pozyskiwanie danych. Oprogramowanie SCADA pozwala komputerowi nadzorować proces technologiczny. Jest to realizowane poprzez pozyskiwanie danych z czujników monitorujących ten proces. Wiele z technik pomiarowych, które zostaną omówione, można uznać za składniki rozwijającej się obecnie technologii zwanej internetem rzeczy (IOT), wykorzystującej otwarte oprogramowanie Node-RED, służące do modelowania procesów.

HMI jest również akronimem (ang. *Human Machine Interface*) i oznacza interfejs człowiek-maszyna (a więc to samo, co panel sterowniczy czy panel operatorski). HMI może być urządzeniem elektronicznym lub konstrukcją, która stanowi interfejs między komputerem, urządzeniem eksperymentalnym, a człowiekiem-operatorem (graficzny interfejs użytkownika, czyli GUI, również może służyć jako HMI).

USB to kolejny akronim (ang. *Universal Serial Bus*), będący w rzeczywistości sformalizowaną specyfikacją, z którą muszą być zgodne systemy sprzętowe. USB jest czymś w rodzaju podsystemu, który pozwala komputerowi osobistemu komunikować się z elementami podłączonymi do uniwersalnej magistrali szeregowej.

Załóżmy, że na komputerze PC działa oprogramowanie SCADA, wykorzystujące interfejs HMI podłączony poprzez USB. W takim przypadku eksperymenty naukowe, procesy, oprzyrządowanie eksperymentalne lub jego konfiguracja mogą być monitorowane i zarządzane w czasie rzeczywistym bez względu na to, czy znajdują się obok komputera, laptopa czy stacji roboczej, czy też są odległe od nich o setki kilometrów.

Laptopy, samodzielne komputery stacjonarne oraz stacje robocze podłączone do internetu poprzez sieć kablową lub bezprzewodową pozwalają obecnie osiągnąć niespotykaną wcześniej uniwersalność podczas monitorowania eksperymentów naukowych, odbywających się w lokalnym laboratorium lub dowolnym miejscu na świecie.

Opcje dostępne dla eksperymentatora, związane z wdrażaniem systemów SCADA, można zasadniczo podzielić na trzy kategorie w zależności od ilości pracy wymaganej do uzyskania w pełni funkcjonalnego środowiska.

Kompletne, działające systemy oprogramowania, które są w stanie mierzyć i kontrolować praktycznie każdy system elektro-optyczno-mechaniczny, są dostępne u takich producentów, jak National Instruments i Foxboro. Ich wersje podstawowe mogą kosztować w granicach kilku tysięcy dolarów.

Autor postanowił w swojej książce zaprezentować trzy znacznie tańsze opcje implementacji systemów SCADA w układach eksperymentalnych.

W trakcie opracowywania ćwiczeń zastosowano umiarkowanie kosztowną strategię implementacji, obejmującą poniższą listę zasobów. Taka lista powinna być również przydatna podczas testowania innych środowisk eksperymentalnych:

1. Komputer PC z oprogramowaniem SCADA. W niniejszej książce wykorzystano DAQFactory Express oraz podstawową wersję systemu DAQFactory firmy AzeoTech (cena podstawowego oprogramowania wynosiła w 2008 roku około 250 dolarów kanadyjskich). Dla osób, które są w stanie dostosować oprogramowanie i mogą potrzebować zwiększonej uniwersalności, producent udostępnił bezpłatne wersje systemów SCADA.
2. Interfejs HMI, podłączony poprzez złącze USB. Również w tym przypadku istnieje wiele urządzeń dostępnych u wielu producentów. W tej książce wykorzystano model U12 firmy LabJack Corporation (jego cena wynosi około 120 dolarów; później dodano również model U3, który kosztuje mniej więcej 110 dolarów). Do urządzeń LabJack jest dołączone oprogramowanie w postaci wersji programu DAQFactory o nazwie „Express”. Ma duże możliwości graficzne i jest więcej niż wystarczające w przypadku wielu zastosowań w technikach śledczych. Wersja DAQFactory Express ma jednak ograniczenia w postaci maksymalnie dziesięciu wierszy kodu skryptowego, pięciu sekwencji skryptowych i dwóch stron wyświetlania. W przypadku niektórych tematów i ćwiczeń projektowych, opisanych w tej książce, może być wymagana komercyjna wersja

oprogramowania DAQFactory, udostępniająca bardziej zaawansowane opcje. Jeśli Czytelnik chciałby kupić taki program, powinien jednak zacząć od najbardziej podstawowej wersji, a rozbudowywać go dopiero w razie konieczności.

3. Trzecią opcją dostępną dla eksperymentatorów jest najnowszy i najtańszy sposób implementacji systemu SCADA, który składa się z urządzenia Raspberry Pi, jego systemu operacyjnego Linux, języka programowania Python z biblioteką *matplotlib* oraz graficznego interfejsu użytkownika *tkinter*. System operacyjny Linux oraz język Python i jego moduły są projektami o otwartych źródłach, a więc można je pobrać z internetu za darmo. Płytkę Raspberry Pi można kupić za około 35 dolarów w wielu sklepach z elektroniką, takich jak DigiKey lub Newark element14. Do urządzenia można podłączyć telewizor zgodny z HDMI lub monitor komputerowy, mysz i klawiaturę, a dzięki temu uzyskać w pełni funkcjonalny system komputerowy. System Raspberry Pi udostępnia oprócz złącza USB magistralę wejścia i wyjścia ogólnego przeznaczenia, a zatem ma w pełni działający interfejs HMI, który może zostać bezpośrednio podłączony do zewnętrznych urządzeń lub środowisk eksperymentalnych. Płytkę Raspberry Pi zawiera również gniazdo Ethernet, więc dzięki temu jest przystosowana do pracy w sieci.

W 2008 roku pojawił się projekt Arduino o otwartych źródłach, który zawierał zestaw płytek z mikrokontrolerem, wykorzystujących złącze USB. Umożliwił on niezliczonym projektantom, artystom, hobbystom i osobom niebędącym specjalistami od elektroniki na podłączanie urządzeń elektro-optyczno-mechanicznych do komputera. Podstawową płytkę Arduino Uno Rev3 można kupić w każdym większym sklepie z elektroniką za około 25 dolarów. System do programowania mikrokontrolera to kolejny projekt o otwartych źródłach, jest więc darmowy i można go pobrać z internetu. Płytkę Arduino może być używana z systemami operacyjnymi Windows lub Linux i jest w pełni wspierana przez społeczność internetową. Zainteresowane osoby mają dostęp do wielu samouczków oraz szerokiego spektrum przykładowych programów użytkowych.

Eksperymenty wykorzystujące systemy typu SCADA mogą się zatem opierać na sprzęcie i oprogramowaniu dostępnym na rynku, możliwym do natychmiastowego użycia w stanie, w jakim je otrzymano, bez konieczności poświęcania czasu na ich rozbudowę. Taki niezbyt kosztowny system wymaga umiarkowanej ilości programowania i wykorzystuje program DAQFactory oraz komercyjne urządzenia HMI,

takie jak interfejsy LabJack. Alternatywnie można też użyć zestawu bardzo taniego sprzętu i otwartego oprogramowania, dostępnego w internecie.

Oprócz oprogramowania i sprzętu, niezbędnego do realizacji systemu monitorującego i sterującego, mogą być wymagane następujące urządzenia pomocnicze:

1. Płytki prototypowa, odpowiednie źródła zasilania, takie jak baterie lub zasilacze regulowane elektronicznie oraz dostęp do różnych układów scalonych i pasywnych elementów elektronicznych.
2. Do rozwiązywania problemów będzie wymagany multimetr, a do bardziej zaawansowanych prac może być potrzebny oscyloskop (jako samodzielne urządzenie lub w postaci programu oscyloskopowego do komputera PC).

Sugeruje się, aby Czytelnik, który jest początkujący w tej technologii, zapoznał się z książką w kolejności rozdziałów, aby nabrać wprawy i pewności siebie w posługiwaniu się oprogramowaniem i okablowaniem oraz realizowaniu coraz bardziej złożonych projektów. Oprogramowanie skryptowe, interfejsy sprzętowe, elektronika i zastosowanie układów scalonych będą w miarę czytania stawały się coraz bardziej skomplikowane. Podstawowa wiedza i procedury, zaprezentowane we wcześniejszych rozdziałach, nie będą przypomniane w późniejszych. Ponieważ cała nauka jest z natury empiryczna, ta książka nie różni się od prawdziwej pracy naukowej. Badacz musi przechodzić od prostych do coraz bardziej skomplikowanych aspektów danego projektu, weryfikując i walidując każdy etap pośredni w wielostopniowym procesie pomiarowym.

Tempo, w jakim dana osoba może przeanalizować zaprezentowane zagadnienia, będzie zależało od jej wiedzy z zakresu podstawowych nauk fizycznych, które stanowią podłoże eksperymentów. W przypadku napotkania trudności można skorzystać z podręczników, samouczków dostępnych w sieci i kursów akademickich z ćwiczeniami, aby zdobyć wymaganą wiedzę podstawową.

Jak wskazuje tytuł, w tej książce zostały przeanalizowane monitorowanie i mierzenie parametrów fizyko-chemicznych za pomocą układów scalonych i sprzętowych systemów obliczeniowych. Do monitorowania i sterowania urządzeń eksperymentalnych są wykorzystywane niedrogie komponenty, które można kupić w sklepach. Zwracają one dane w postaci podstawowych parametrów fizyko-chemicznych, przydatnych dla badaczy z wielu dziedzin naukowych. W niektórych przypadkach uzyskano zdumiewającą czułość, uniwersalność, dokładność i precyzję testowanych systemów.

Ostrzeżenia i wyłączenie odpowiedzialności

1. Prąd o napięciu 230 V może być śmiertelnie niebezpieczny i potrafi spowodować pożar.
2. Lutownice są wystarczająco gorące, aby spowodować poważne poparzenia.
3. Niniejsza praca służy wyłącznie celom edukacyjnym i prezentuje koncepcje, które zostały zaimplementowane w postaci eksperymentalnej. Nie zostały one przetestowane pod kątem wytrzymałości ani zaprojektowane czy też przeznaczone do wdrożenia produkcyjnego. Koncepcje te stanowią jedynie podstawę edukacyjną i są punktami wyjściowymi do dalszych badań i rozwoju w zakresie instrumentalnych metod monitorowania eksperymentalnej aparatury naukowej w celu gromadzenia danych lub wykonywania pomiarów fizycznych.
4. Pomysł na tę książkę pojawił się u autora w połowie lat 60. XX wieku, a w latach późniejszych różne jej fragmenty były tworzone przy użyciu dostępnych wówczas technologii, podczas gdy pewne koncepcje okazywały się niewykonalne. Chociaż formalne przeprowadzanie eksperymentów rozpoczęło się w latach 2008-2009 przy użyciu dostępnych wówczas układów scalonych, oprogramowania do obliczeń fizycznych i internetowych zasobów informacyjnych, dokument w trakcie tworzenia był modyfikowany ze względu na użycie nowszych układów scalonych, odpowiedniego oprogramowania i wiedzy dostępnej w internecie. Ponieważ nigdy nie może zostać zapewniona ciągła dostępność oprogramowania lub sprzętu, praktycy zajmujący się jakąkolwiek dziedziną nauki muszą się nauczyć sztuki tworzenia rozwiązań zastępczych.



Sposób wykonywania ćwiczeń

Jak wspomniano wcześniej, autor nie chciał stworzyć książki o podstawach gromadzenia danych, przeznaczonej dla początkujących Czytelników. Chociaż zmotywani lub entuzjastycznie nastawieni badacze mogliby od razu przystąpić do pracy i w jej trakcie próbować zdobywać potrzebną wiedzę i umiejętności, przewodnik ten jest skierowany do osób, które mają przynajmniej pewne doświadczenie w obsłudze sprzętu elektronicznego i oprogramowania komputerowego. Podstawowa znajomość prostej elektroniki, jak również elementarnego programowania w języku strukturalnym, takim jak Python lub C++, skróci czas potrzebny do wykonania poszczególnych ćwiczeń.

Dokumentacja dostarczana przez producentów większości urządzeń do gromadzenia danych, o których będzie mowa w niniejszej książce, zawiera wskazówki i opisy podstawowych czynności, które pomagają zapoznać nowego użytkownika z ich obsługą. Źródła internetowe również dostarczają wielu praktycznych informacji.

Po opanowaniu przez eksperymentatora sprzętu i oprogramowania opisanych w tej pracy, może on zacząć przeprowadzać pomiary wielu podstawowych parametrów fizycznych zgodnie z metodami opisanymi w następnej książce z tej serii (*Arduino Measurements in Science*).

Ta pozycja dotyczy opracowywania metod, które mogą zostać wykorzystane do przeprowadzania eksperymentalnych pomiarów fizykochemicznych za pomocą sprzętu utworzonego z łatwo dostępnych elementów, materiałów i prostych, stacjonarnych lub przenośnych systemów komputerowych. Jest ona próbą przedstawienia metodologii, dzięki której podstawowe pomiary mogą być wykonywane przez badaczy o różnym poziomie znajomości elektroniki, elektrooptyki i nieskomplikowanych układów mechanicznych. Zestaw eksperymentalnych procedur pomiarowych jest wstępem do możliwości wykonywania pomiarów takich parametrów, jak temperatura, odległość, natężenie światła, częstotliwość dźwięku, wilgotność względna i inne wielkości, wykorzystywane w naukach podstawowych.

W każdym z rozdziałów jest prezentowana metoda lub technika, która może zostać ostatecznie wykorzystana podczas opracowywania metody testowej lub pomiarowej, a także procedury składającej się z różnych zaprezentowanych metodologii.

Składniki ćwiczeń

- ◆ Eksperyment.
- ◆ Sprzęt.
- ◆ Oprogramowanie.
- ◆ Przebieg eksperymentu.
- ◆ Analiza.
- ◆ Listingi kodów.

Zarządzanie projektem

Podczas przeprowadzania ćwiczeń w poszczególnych rozdziałach warto przestrzegać następujących zaleceń:

1. Przygotowując się do montażu eksperymentalnego urządzenia lub projektu opisanego w rozdziale, należy przeczytać informacje zawarte w książce i zebrać materiały istotne dla ćwiczenia, takie jak karty katalogowe używanych komponentów. Pozwoli to na pogłębienie wiedzy i może zapobiec uszkodzeniu elementów. Wstępne notatki i rysunki powinny zostać zebrane w jednym miejscu (na papierze lub w formacie elektronicznym).
2. Montaż sprzętu/elektroniki i projektowanie odpowiedniego oprogramowania powinno się rozpoczynać od utworzenia najprostszych elementów składowych, testować poszczególne moduły, a następnie weryfikować stan systemu, aż cały projekt będzie funkcjonował zgodnie z założeniami.
3. Należy zachować ostrożność podczas czytania schematów i próbach ich implementacji, ponieważ niektóre elementy dyskretne i układy scalone mają coraz mniejsze rozmiary fizyczne lub są zastępowane przez nowsze technologie. Zmniejszanie rozmiarów oznacza, że oznaczenia identyfikacyjne na komponentach również stają się coraz mniejsze.

Oznaczenia na rezystorach oraz kondensatorach mogą mieć różne formaty i występować w postaci kombinacji cyfr i liter z symbolem rzędu wielkości zastępującym czasem przecinek dziesiętny. Elementy do montażu powierzchniowego (SMT) mają

kod trzycyfrowy, w którym pierwsze dwie cyfry oznaczają wartość, a trzecia jest potęgą dziesiątą mnożnika.

Jednostką oporności jest om (symbol Ω).

Litera M oznacza wartość 10^6 lub 1 000 000 Ω . Typowymi wielkościami oporności mogą być 1,5M lub 1M5.

Litera K oznacza wartość 10^3 lub 1 000 Ω . Typowymi wielkościami oporności mogą być 1,2K lub 1K2.

Litera R oznacza 10^0 lub 1 Ω . Typowymi wielkościami oporności mogą być 100R lub po prostu 100, ponieważ w takich przypadkach nie wykorzystuje się przecinka dziesiątą.

Litera m oznacza 10^{-3} lub 1/1000 Ω , więc wielość 0,052 Ω zapisuje się jako 52 m Ω .

Jednostki pojemności, używane w starszych książkach, były ograniczone głównie do mikro- i pikofaradów. Zakres nano- był uzyskiwany za pomocą tysięcy pikofaradów, a zakres mili- za pomocą tysięcy mikrofaradów. Większość obecnych notacji dla kondensatorów wydaje się wykorzystywać trzy główne zakresy wymienione poniżej, ale ostatnio rozszerzono je o całe farady, aby uniknąć używania tysięcy i milionów mniejszych jednostek przy opisywaniu ultra- i superkondensatorów.

Jednostką pojemności jest farad (symbol F).

Litera μ oznacza mikrofarad, który jest równy 10^{-6} farada.

Litera n oznacza nanofarad, który jest równy 10^{-9} farada.

Litera p oznacza pikofarad, który jest równy 10^{-12} farada.

Ćwiczenia zaprezentowane w tej książce wykorzystują bardzo proste układy elektroniczne, które są montowane na płytce prototypowej i podłączane do interfejsu HMI LabJack, systemu DAQFactory Express, zestawu mikrokontroler Arduino-DAQFactory lub bezpośrednio do systemów Raspberry Pi lub zestawu Raspberry Pi-Arduino. Dzięki temu uzyskuje się działający obwód elektroniczny z komputerowo wygenerowanym graficznym interfejsem użytkownika. Każda z tych kombinacji pozwala eksperymentatorowi na nadzorowanie procesu, pozyskiwanie danych lub monitorowanie ich strumienia poprzez oprogramowanie wyświetlające informacje na ekranie. Przed skonfigurowaniem oprogramowania do pozyskiwania danych (DAQ) i rozpoczęciem obsługi sprzętu nie ma lepszego sposobu na zdobycie doświadczenia z elektromechanicznymi systemami sterowania od fizycznego zmontowania obwodów, przetestowania i sprawdzenia ich działania. Ogólna zasada polega

na tym, by sprzęt był montowany, testowany i walidowany, zanim przejdzie się do tworzenia interfejsów i oprogramowania.

W poniższej analizie, na podstawie pierwszego z ćwiczeń zaprezentowano ogólną metodologię, która będzie stosowana w pozostałych. Każde ćwiczenie w tej książce zostaje najpierw ogólnie przedstawione w sposób typowy dla badań laboratoryjnych. Założono, że to omówienie zostanie przeczytane przez badacza i będzie przez niego zrozumiałe.

Podczas pracy z sygnałami elektrycznymi, pochodzącymi z czujnika lub aparatury eksperymentalnej, należy się upewnić, że poziom napięcia wyjściowego nie przekracza bezpiecznego zakresu napięcia wejściowego elementów elektronicznych, używanych do przetwarzania takich sygnałów. Większość dyskretnych układów scalonych używa napięcia o maksymalnej wartości 5 V, niektóre wzmacniacze operacyjne mogą pracować przy napięciu do 18 V, a znaczna część elementów wykorzystujących technologię montażu powierzchniowego działa przy nominalnym napięciu 3,3 V.

Podobnie jak w przypadku wszystkich przedsięwzięć naukowych, należy zaczynać od najprostszych konfiguracji i w miarę postępu zajmować się bardziej złożonymi. Opracowując oprogramowanie dla danego projektu eksperymentator powinien zacząć od kodu wymaganego do połączenia aparatury z układem obliczeniowym i wyświetlającym.

Najprostsza forma transmisji sygnału elektrycznego wykorzystuje połączenie szeregowo zarówno w przypadku analogowym, jak i cyfrowym.

Napięciowe sygnały analogowe są często podawane na takie wyprowadzenia wejściowe, które umożliwiają ich przetwarzanie, natomiast sygnały cyfrowe są dostarczane do wyprowadzeń rozpoznających, czy panuje na nich stan wysoki lub niski. Ogólnie rzecz ujmując, większość sygnałów wyjściowych, uzyskiwanych z czujników jest napięciowych, ale w niektórych przypadkach stosuje się również natężenie prądu.

Systemy obliczeniowe zazwyczaj uzyskują dostęp do danych zewnętrznych poprzez tak zwany „port szeregowy”. Port szeregowy jest często określonym, adresowalnym miejscem w pamięci komputera, w którym zgodnie z określonym kodowaniem zwanym protokołem są zapisywane nadchodzące dane cyfrowe. Protokół określa znaczenie przejść ze stanu wysokiego do niskiego (lub odwrotnie), które tworzą sygnał cyfrowy zmieniający się w funkcji czasu, odpowiednie wartości danych i parametry sterujące ich przetwarzaniem. Istnieje wiele naukowych i przemysłowych protokołów transmisji szeregowo, zaprojektowanych i zoptymalizowanych dla określonych zastosowań, jednak ćwiczenia zaprezentowane w tej książce będą w większości wykorzystywać jedynie podstawy szeregowo transmisji danych.

W ćwiczeniach może być używany język skryptowy DAQFactory, Python oraz odmiana języka C używana w programowaniu Arduino. Te trzy języki programowania mają zarezerwowane pewne słowa kluczowe, które nie mogą być stosowane jako nazwy zmiennych. Zawsze należy postępować zgodnie z zasadami nazewnictwa zmiennych, podanymi w dokumentacji odpowiedniego języka. Powinno się używać znaczących nazw i stosować tradycyjne style nazewnicze, na przykład `MySignificantName`, `MySgnfcntNme` lub `My_Significant_Name`. W kodzie Pythona, skryptach DAQFactory, językach C lub C++ nie należy używać pełnych słów pochodzących ze słownika języka angielskiego, takich jak `temperature` czy `Temperature`. Okazuje się, że skrypty, w których dla ułatwienia użyto jako nazw zmiennych lub kanałów poprawnych słów angielskich, czasem nie działały i zwracały zaskakujące wyniki. Gdy w takich nazwach zostały użyte wielkie i małe litery, skrypty nagle zaczynały działać bez zarzutu. Należy więc postępować zgodnie z właściwą metodologią, wynikającą ze standardu danego języka programowania. Jeśli w przypadku DAQFactory zostaną najpierw utworzone kanały, pozwoli to na używanie rozwijanych menu z ich listami, a także nazwami zmiennych i stałych, a dzięki temu zmniejszeniu ulegnie liczba błędów pojawiających się podczas wprowadzania kodu. Pierwszym krokiem podczas rozwiązywania problemów z programem, który nie działa, jest sprawdzenie jego poprawności. Należy pamiętać o tym, że nazwy zmiennych uwzględniają wielkość liter.

Powinno się także prowadzić szczegółowe notatki z wykonywanych czynności, zapisywać obliczenia, szkicować schematy i wstępne rysunki konstrukcji. Na tym polega nauka. Konwencje rysunkowe dla systemów mechanicznych i obwodów elektronicznych można znaleźć w wielu materiałach źródłowych*. Warto ich przestrzegać.

Podczas tworzenia połączeń na płytce prototypowej należy testować każdy składnik układu. Sama praca powinna być wykonywana starannie, a przewody muszą być układane równoległe do rzędów wyprowadzeń, dostępnych na płytce drukowanej. Przewody należy krzyżować jedynie pod kątem prostym. Powinny być zaginane w taki sposób, aby uzyskać stosunkowo duży promień krzywizny. Pamiętajmy, że miedź, choć bardzo plastyczna, twardnieje podczas wyginania, więc w miarę możliwości na-

* 1. Moore, Davis i Coplan, *Building Scientific Apparatus*, wydanie IV, Cambridge University Press, ISBN 978-0-521-8785-6
2. Horowitz i Hill, *The Art of Electronics*, wydanie II, Cambridge University Press, ISBN 0-521-37095-7
3. Scherz i Monk, *Practical Electronics for Inventors*, wydanie III, McGraw Hill, ISBN 978-0-07-177133-7.

leży używać nowego drutu lub upewnić się, że będzie on ponownie zginany jedynie kilka razy, by uzyskać łagodne krzywizny o dużym promieniu. W inżynierii chemicznej proces produkcyjny składa się z szeregu „operacji jednostkowych”. Każda operacja jednostkowa jest zazwyczaj kompletnym krokiem, obejmującym zmianę fizyczną lub przemianę chemiczną, taką jak krystalizacja lub wytrącanie, który składa się na większy, wieloetapowy proces produkcyjny. Koncepcja operacji jednostkowych może być zastosowana do utworzenia podstawowego systemu nadzorowania procesu technologicznego i pozyskiwania danych (SCADA). W zasadzie można uznać, że każdy system SCADA składa się co najmniej z trzech elementów: procesu, który ma być kontrolowany, mechanizmu wykrywania i regulacji, a także centralnego zespołu sterującego.

Aby w pierwszym ćwiczeniu zaprezentować koncepcję operacji jednostkowych, należy odpowiednio skonfigurować program DAQFactory w celu aktywowania kanału. Zostanie mu przypisane odpowiednie wyjście z urządzenia LabJack. To urządzenie będzie z kolei podłączone do wejścia układu sterownika. Wyjścia sterownika prądowego zostaną podłączone do rezystora ograniczającego prąd przepływający przez diody LED, używane przez układ.

Pierwszy krok w procedurze testowej polega na sprawdzeniu za pomocą woltomierza cyfrowego napięć $+5$ i 0 V na wyjściowym wyprowadzeniu LabJack. Takie napięcia powinny zostać zmierzone na wyprowadzeniu 3 buforowego sterownika prądowego CD4050 oraz na końcówce rezystora ograniczającego prąd diod LED, odpowiadającej wyższemu potencjałowi. Przyjmuje się, że jeśli wszystkie części składowe systemu działają, cały proces będzie również działać. Należy jednak pamiętać, że jest to tylko założenie!

Izolacja

USB jest w zasadzie standardem komunikacyjnym, więc z tego powodu ma ograniczone zdolności dostarczania zasilania. W ćwiczeniu wykorzystano interfejs HMI LabJack, który jest w rzeczywistości zasilany z komputera. LabJack może być też źródłem prądu o natężeniu do 450 mA. W przypadku urządzeń eksperymentalnych dobrym zwyczajem jest stosowanie zewnętrznego zasilacza. Ponieważ wszystkie ćwiczenia będą wykonywane na stole lub biurku, dlatego też podobnie jak w każdym eksperymencie przeprowadzanym w laboratorium, zostanie wykorzystany oddzielny zasilacz. Niektóre urządzenia eksperymentalne mogą pobierać więcej niż pół am-

pera, a inne potrafią sterować napięciem i prądem zasilania. Zasilanie poprzez sieć komputerową lub bezpośrednio z laptopa może nie być wystarczające dla urządzeń eksperymentalnych, zarządzanych za pomocą systemu SCADA. Do zasilania diody LED w pierwszym ćwiczeniu zostanie użyty interfejs HMI, który będzie sterować obwodem „buforowym” układu scalonego CMOS CD4050. Ten z kolei zostanie wykorzystany do włączania i wyłączenia diody LED. Logika sterowania zerojedynkowej, dostępna w oprogramowaniu SCADA i zaimplementowana na wyprowadzeniach wejścia i wyjścia urządzenia HMI w postaci napięć +5 lub 0 V, jest zatem wykorzystywana do sterowania prądem z zewnętrznego zasilacza 6 V.

Do pomiaru sygnałów analogowych o niskim poziomie jest często niezbędna zewnętrzna bateria lub odpowiednio regulowane zasilanie. Badacze korzystający z zasilania USB o napięciu 5 V często stwierdzają, że niskopoziomowe sygnały analogowe są zakłócanie szumem zegara cyfrowego magistrali.

W większości rzeczywistych zastosowań systemy monitorowane i sterowane mają własne zasilanie, dlatego mogą być połączone z komputerem i oprogramowaniem SCADA poprzez łącze bezprzewodowe. Gdy do zasilania jest używane złącze USB, nie należy zbytnio obciążać komputera, a z magistrali pobierać tylko minimalny, wymagany prąd.

Niektóre z konfiguracji eksperymentalnych, które zostaną przetestowane w tej książce, będą pobierać prąd rzędu amperów, dlatego nie mogą być zasilane z komputera, lecz przy użyciu oddzielnego zasilacza. W późniejszych ćwiczeniach będą wykorzystywane mikroprocesory podłączone do USB. Chociaż mogą być one zasilane przez to złącze, powinny korzystać z oddzielnego zasilacza, aby zapobiec nakładaniu się szumów z systemu magistrali na sygnał danych na wyjściu mikroprocesora (złącza USB w wersji 1 i 2 umożliwiają uzyskanie prądu zasilającego w wysokości 500 mA, a USB w wersji 3 900 mA).

Skrypty programowe

Każdy gotowy skrypt powinien zostać w pełni udokumentowany. Wykorzystując zasady nazewnictwa i komentowania, specyficzne dla danego języka programowania, w nagłówku pliku należy umieścić nazwę sekwencji* lub kodu, cel programu i ewen-

* Jak wyjaśniono w dokumentacji DAQFactory, sekwencja jest czymś w rodzaju programu skryptowego (zestawu instrukcji), który umożliwia wykonywanie określonych działań – *przyp. tłum.*

tualnie datę utworzenia. Nagłówek powinien także zawierać opis działania kodu, algorytmu i objaśnienia dotyczące używanych zmiennych. Należy też pamiętać, że zmienne muszą zostać zadeklarowane. Oprócz tego skrypt powinien zostać uruchomiony, aby zmienna została utworzona i mogła być używana. System DAQFactory posiada opcję automatycznego uruchamiania skryptów. Jest ona aktywowana, gdy zostanie wyświetlony ekran kontrolny, któremu został przypisany określony skrypt. Dzięki temu można automatycznie zadeklarować i utworzyć zestaw zmiennych, które następnie będą używane na ekranie kontrolnym lub w sekwencji.

Systemy Raspberry Pi i Arduino automatycznie uruchamiają swoje systemy operacyjne i definiują zmienne programowe po włączeniu zasilania.

Układy scalone i technologia montażu powierzchniowego

Kiedyś eksperymenciści podczas testowania i tworzenia projektów kupowali komponenty przeznaczone do montażu na płytkach prototypowych. Układ działający na takiej płytce uniwersalnej mógł zostać następnie przekształcony na jednostronną lub dwustronną płytkę drukowaną. W płytkach dwustronnych często istniały otwory, przez które były łączone ścieżki górne i dolne. Ponieważ jednak układy scalone stały się znacznie mniejsze i pobierają mniej prądu, są więc szybsze, ale jednocześnie dużo bardziej wrażliwe na pole elektrostatyczne. Wiele z nich nie może być tak jak wcześniej ręcznie montowanych przez typowego badacza-eksperymentatora.

Mniejsze rozmiary układów scalonych przyczyniły się do zmniejszenia komponentów i powstania technologii montażu powierzchniowego (ang. *surface-mount technology*, w skrócie SMT), która z kolei sprawiła, że również płytki drukowane stały się znacznie mniejsze, łatwiejsze w produkcji i tańsze. Małe rozmiary elementów i rozwój technologii SMT dodatkowo skomplikowały życie eksperymentatora. Wykorzystanie zalet, wynikających z fizycznego zmniejszenia rozmiarów układów scalonych, wymaga zastosowania adapterów do konwersji komponentów SMT na formaty kompatybilne z rozmiarami otworów w płytkach prototypowych.

W ćwiczeniach z kolejnych rozdziałów zostaną wykorzystane przede wszystkim łatwo dostępne układy scalone, które są kompatybilne z powszechnie stosowanymi systemami płytek uniwersalnych.

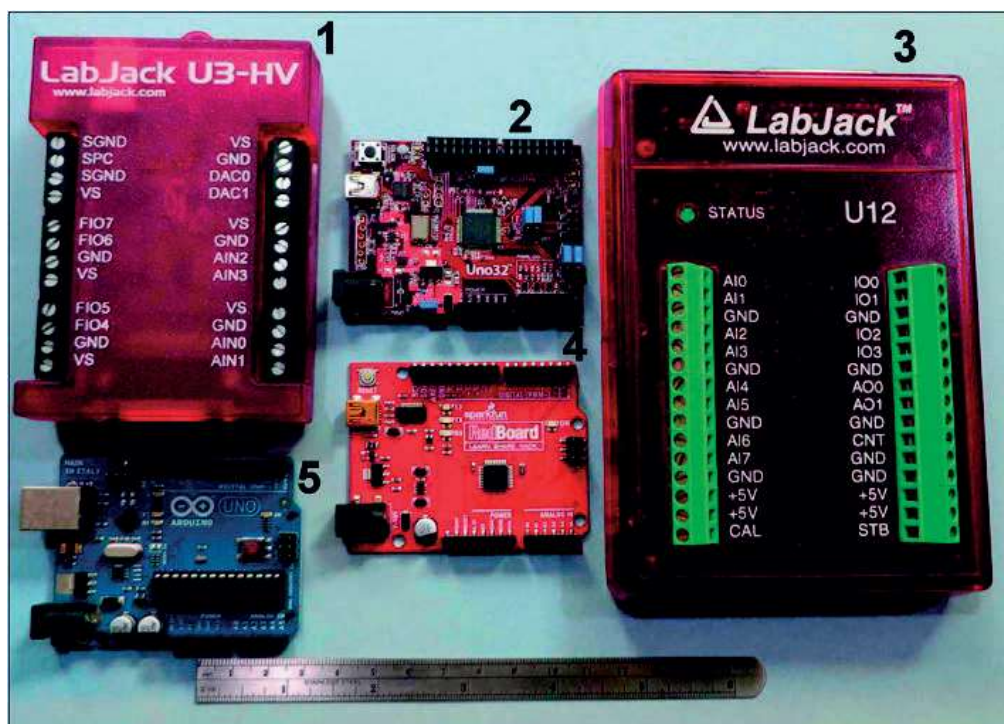
Włączanie i wyłączanie diody LED za pomocą przycisku

Ćwiczenie w tym rozdziale jest ogólnie rzecz ujmując jedną z najprostszych form sterowania komputerem. Polega ono na włączaniu i wyłączaniu diody LED poprzez kliknięcie przycisku na ekranie lub uruchomienie kilku wierszy kodu komputerowego. Na ekranie jest wyświetlany graficzny interfejs użytkownika (GUI). Składa się on z ikon przycisków, suwaków, wskaźników, cyfrowych wyświetlaczy numerycznych, graficznych wykresów słupkowych i innych symboli, zarówno aktywnych/pasywnych, jak i tekstowych, które mogą być używane do monitorowania i sterowania danego procesu. Kliknięcie przycisku na ekranie powoduje włączenie lub wyłączenie diody LED, a stan systemu jest rozpoznawany wizualnie poprzez stwierdzenie, czy element świeci. Pierwsze ćwiczenie, umożliwiające połączenie oprogramowania SCADA z interfejsem HMI i układem elektronicznym na płytce prototypowej, pozwala więc sterować zasilaniem prostych urządzeń za pomocą ekranu lub klawiatury.

W celu połączenia oprogramowania graficznego, działającego w systemie operacyjnym Windows z płytką przeznaczoną do szybkiego prototypowania, umieszczoną na stole warsztatowym lub wbudowaną w środowisko eksperymentalne, wymagane jest łącze cyfrowe. Jako linia transmisyjna sygnału elektrycznego, łącząca komputer z interfejsem maszyny, może zostać wykorzystany kabel USB, używany do podłączania urządzeń peryferyjnych.

Interfejs maszyny może stanowić dowolne z wielu programowalnych urządzeń, kompatybilnych z USB, zdolnych do odbierania kodu cyfrowego, jego interpretowania oraz generowania wymaganej odpowiedzi.

Na rysunku 1.1 przedstawiono wybrane urządzenia, służące do pozyskiwania danych i działające jako interfejsy HMI, które są w stanie zapewnić wymaganą interpretację danych wejściowych i wygenerować odpowiedni sygnał wyjściowy.



Rysunek 1.1 Przykłady urządzeń HMI

Urządzenia oznaczone cyframi 1 i 3 pochodzą z amerykańskiej firmy LabJack Corporation, mającej siedzibę w Lakewood (Kolorado). Korporacja ta produkuje kilkanaście wielofunkcyjnych urządzeń do pozyskiwania danych (DAQ), kompatybilnych z systemami USB, Ethernet i Wi-Fi. Jest to wytrzymały i solidny sprzęt, przeznaczony do wymagających zastosowań przemysłowych i laboratoryjnych, charakteryzujący się mocnymi, plastikowymi obudowami ochronnymi i dużymi zaciskami śrubowymi, służącymi do podłączania przewodów. Dwa przedstawione urządzenia to najtańszy model U3-HV (w cenie 115 dolarów; opis dostępnych funkcji i możliwości jest dostępny w dołączonej dokumentacji) oraz oryginalny, wielofunkcyjny przyrząd LabJack do pozyskiwania danych (w cenie 160 dolarów).

Urządzenie o numerze 2 to mikrokontroler chipKIT Uno32 firmy Digilent Inc. (wycofany z produkcji), kompatybilny z Arduino. Przedstawiony układ został zastąpiony elementem chipKIT uC32 – 3,3-woltowym mikrokontrolerem kompatybilnym z Arduino (w cenie 42 dolarów kanadyjskich). Pozycje 4. i 5. to niedrogie,

masowo produkowane mikrokontrolery firmy SparkFun Inc. (20-30 dolarów), dostępne w technologii SMT i kompatybilne z Arduino.

Każde z pokazanych urządzeń jest w stanie odebrać pojedynczy, włączająco/wyłączający sygnał cyfrowy, a także zakodowaną instrukcję, a następnie wygenerować wymagane dane wyjściowe.

Eksperyment

Pierwsza wersja systemu sterującego jasnością diody LED za pomocą przycisku została utworzona w 2007 roku na niezależnie zasilanej płytce prototypowej. Projekt wykorzystuje urządzenie LabJack U12 (rysunek 1.1, element o numerze 3). Wiele lat później, dzięki dostępności tanich płytek z mikrokontrolerami, do realizacji tego samego zadania mogły zostać wykorzystane urządzenia o numerach 2, 4 i 5, przedstawione na rysunku 1.1. Mimo że w tym ćwiczeniu zastosowano przyrząd U12, jako jego zamiennik może zostać użyta każda płytka z mikrokontrolerem. Szczegóły dotyczące konfiguracji przycisku na ekranie kontrolnym systemu DAQFactory, który wykorzystuje środowisko Arduino do włączania diody LED, zostały podane w rozdziale 11. Opisano w nim, w jaki sposób można skonfigurować trzy przyciski, aby włączać oraz wyłączać diodę LED za pomocą zaprogramowanej sekwencji.

Sprzęt

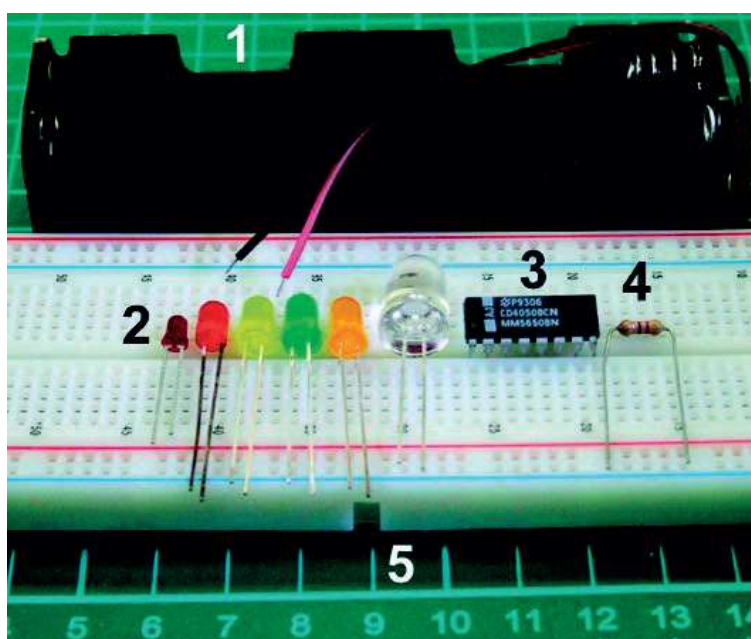
Rysunek 1.2 przedstawia typowe elementy elektroniczne, wykorzystywane w tym ćwiczeniu. Zostały one ułożone na płytce prototypowej jeden obok drugiego.

Element o numerze 1 jest plastikową obudową na baterie, która może pomieścić osiem ogniw rozmiaru AA i zapewnić napięcie nominalne 12 V. Jest to prawdą jedynie w przypadku nowych ogniw alkalicznych. Osiem ogniw niklowo-metalowo-wodorkowych (NiMH) 1,2 V pozwoli na uzyskanie napięcia 9,6 V. Istnieją też ogniwa litowe rozmiaru AA, które charakteryzują się napięciem nominalnym równym 3,6 V.

Elementy przy cyfrze 2 są diodami LED. Czerwona dioda LED po lewej stronie jest miniaturowym urządzeniem o średnicy 3 mm. Kolejne cztery kolorowe diody mają średnicę 6 mm, natomiast przezroczysty element po prawej stronie jest białą diodą LED o średnicy 10 mm.

Element numer 3 to układ scalony CD4050 zawierający sześć buforów nieodwracających (dokładniej pokazany na rysunku 1.3). Element 4. jest typowym rezystorem

o mocy 1/8 wata, służącym do ograniczania prądu, natomiast element 5. płytką prototypową o wymiarach 50 mm na 163 mm. Posiada ona w górnej i dolnej części dwie niezależne linie zasilające, oznaczone kolorami czerwonym (+) i niebieskim (-). Do każdej takiej linii może zostać podłączone 50 punktów zasilania i 50 punktów uziemienia. Pomędzy górnymi a dolnymi liniami zasilania znajdują się dwie niezależne grupy składające się z 63 kolumn, z których każda zawiera pięć punktów połączeniowych.

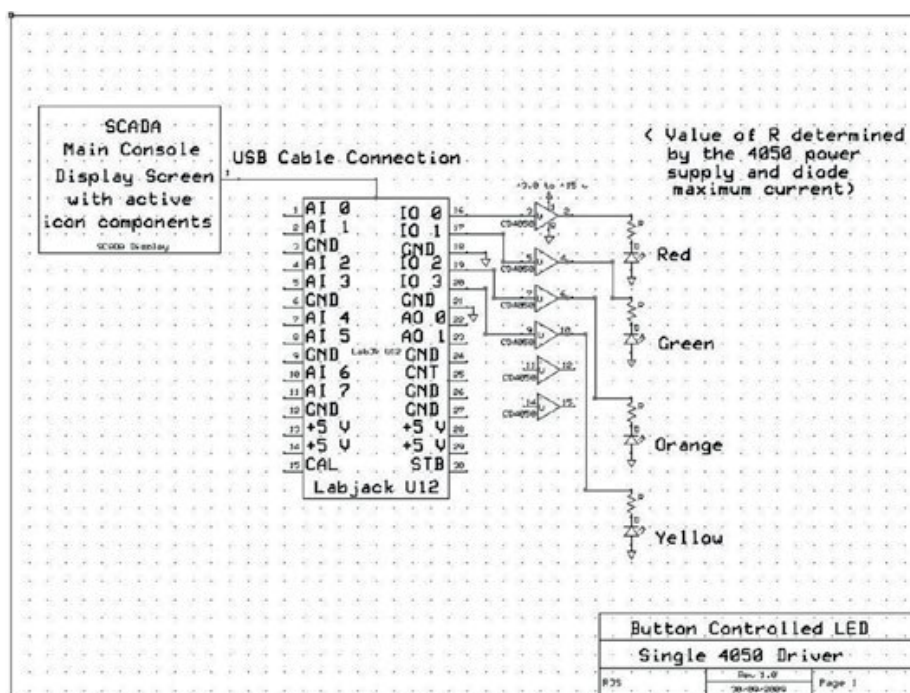


Rysunek 1.2 Płytką prototypową i używane elementy

Schemat układu: połączenie systemu DAQFactory i urządzenia LabJack

Schemat przedstawiony na rysunku 1.3 będzie wykorzystywany w pierwszych dwóch ćwiczeniach. Cały układ, składający się z czterech diod LED, zostanie użyty w rozdziale 2. do monitorowania zużycia energii przez poszczególne elementy świejące. W pierwszym ćwiczeniu należy użyć połączenia dotyczącego pojedynczej, czerwonej diody LED.

Poszczególne przełączniki, dostępne w układzie CD4050 są jednokierunkowe – zmiana napięcia przyłożonego do wejścia wpływa na sygnał wyjściowy, ale wahania napięcia na wyjściu nie mają wpływu na wejście.



Rysunek 1.3 Schemat układu sterowania poborem mocy przez diody LED

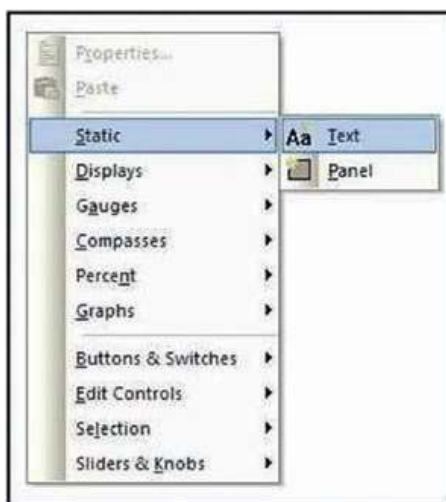
Oprogramowanie

Wymagany komponent: pojedynczy przycisk

Jak wspomniano już w podrozdziale „Sposób wykonywania ćwiczeń”, w przypadku prac związanych z dowolną dyscypliną naukową powinno się zawsze prowadzić szczegółowe notatki. W tym ćwiczeniu należy wybrać nazwę kanału, który ma zostać użyty do sterowania interfejsem HMI, czyli w tym przypadku urządzeniem LabJack. Następnie trzeba skonfigurować oprogramowanie w taki sposób, aby dostarczało sygnał wyjściowy kanału do pierwszego zacisku wejścia i wyjścia (I/O 0), dostępnego na płycie zaciskowej urządzenia (więcej szczegółów na ten temat w instrukcji obsługi LabJack U12 lub U3-HV). Zawsze należy postępować zgodnie z dokumentacją użytkownika, dostarczoną z systemami DAQFactory i LabJack. W tym ćwiczeniu użyto nazwy kanału *RedLed*. Kanały, które mają być używane w dowolnym projekcie DAQFactory, powinny zostać skonfigurowane i uaktywnione przyciskiem *Apply* (zastosuj) przed umieszczeniem elementów na ekranie. Dzięki temu nazwy kanałów pojawią się w menu rozwijanym, jak widać na rysunku 1.11 (jeśli Czytelnik chciałby użyć mikrokontrolera zamiast urządzenia LabJack, powinien się zapoznać z rozdziałem 11.).

W tym ćwiczeniu, podobnie jak w przypadku każdego tworzonego oprogramowania, należy zadbać o dokumentację. Niechęć do dokumentowania programów jest w najlepszym wypadku złym nawykiem. Przed umieszczeniem i skonfigurowaniem przycisku powinno się umieścić na ekranie informację tekstową, aby opisać jego funkcję. Pole tekstowe jest tworzone za pomocą menu podręcznego, wyświetlanego po kliknięciu prawym przyciskiem myszy. Z menu należy wybrać opcję *Static* (element statyczny), a następnie *Text* (tekst). Spowoduje to pojawienie się okna dialogowego, umożliwiającego utworzenie komunikatu ekranowego.

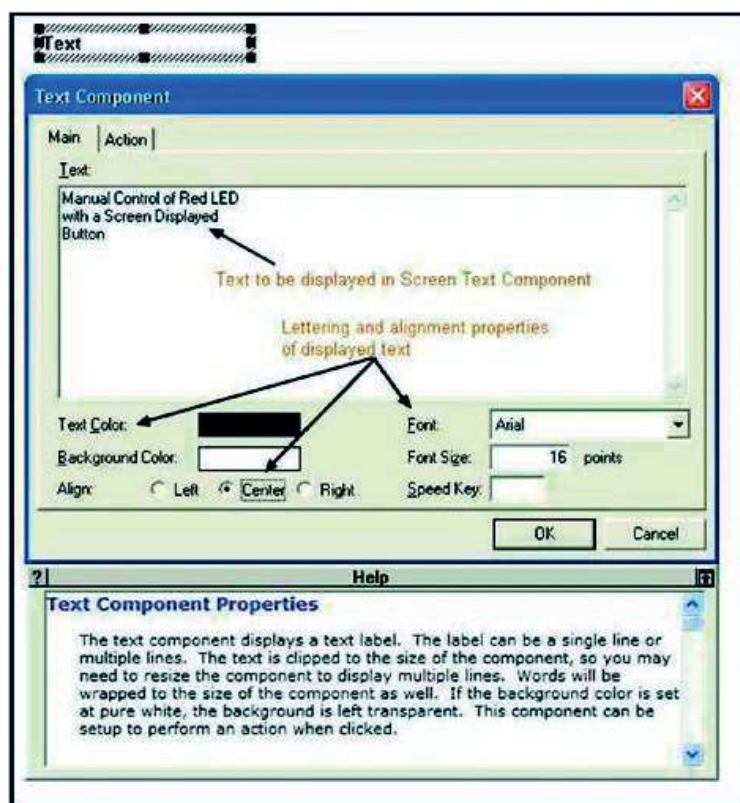
Rysunki 1.4, 1.5 i 1.6 przedstawiają wybór statycznego komponentu tekstowego, wyświetlenie menu podręcznego z opcjami komponentu oraz aktywny panel wprowadzania tekstu.



Rysunek 1.4 Kontrolka prostego przycisku



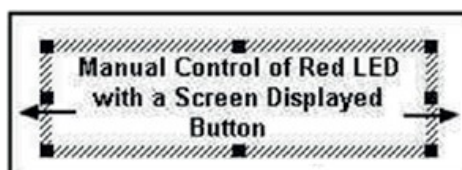
Rysunek 1.5 Właściwości prostego przycisku



Rysunek 1.6 Konfiguracja pola tekstowego

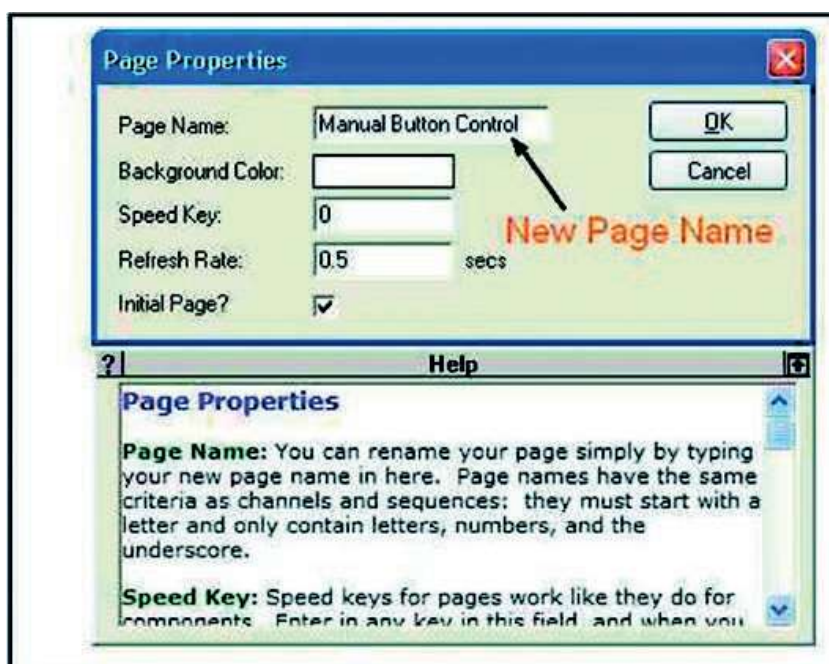
Po umieszczeniu ramki tekstu we właściwym miejscu na ekranie, należy ją kliknąć prawym przyciskiem myszy, co sprawi, że zostanie wyświetlone menu zawierające między innymi opcję *Properties* (właściwości). Jej wybranie otworzy okno właściwości, umożliwiające wprowadzenie napisu, który powinien zostać wyświetlony w komponentie tekstowym.

Po wprowadzeniu żądanej treści do pola tekstowego i wybraniu odpowiednich opcji związanych z wyrównaniem tekstu, kolorem, rodzajem i rozmiarem czcionki, można przyciskiem *OK* zamknąć okno dialogowe, aby umieścić kontrolkę tekstową na ekranie. Do wyświetlenia całej wiadomości może być konieczne rozszerzenie (powiększenie) pola tekstowego, widocznego na rysunku 1.7.



Rysunek 1.7 Zmiana rozmiaru pola tekstowego

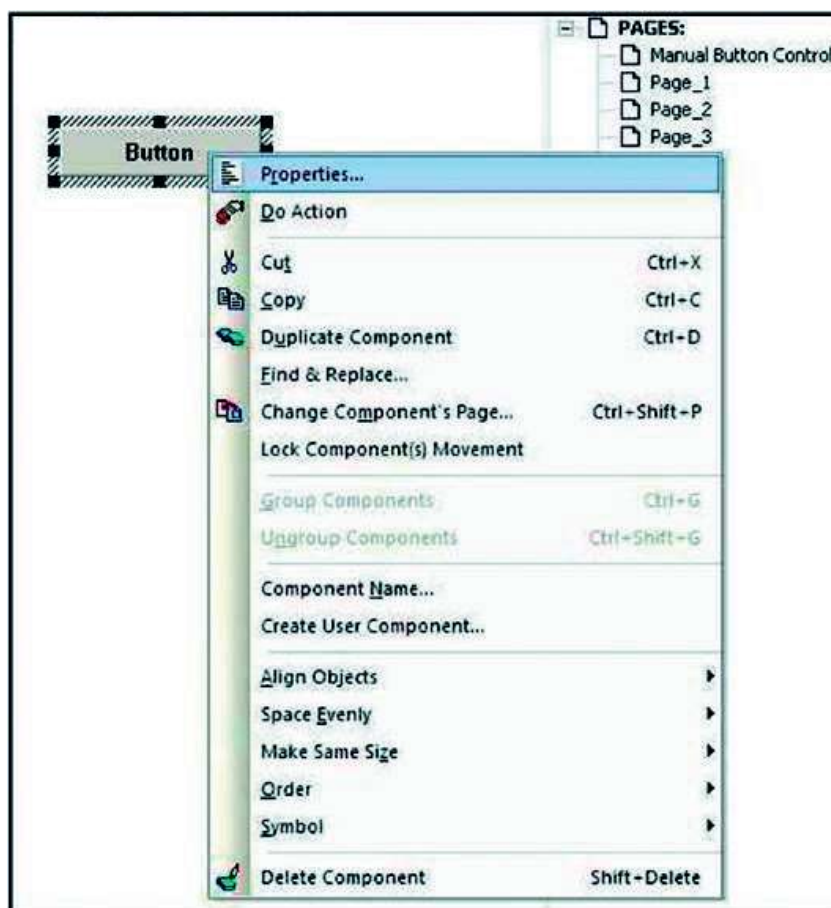
Teraz, zgodnie z filozofią nieustannego dokumentowania, warto odpowiednio nazwać stronę za pomocą okna *Page Properties* (właściwości strony), widocznego na rysunku 1.8. Okno to wyświetli się po kliknięciu prawym przyciskiem myszy bieżącego identyfikatora *page_n*, dostępnego na liście stron, a następnie wybraniu środkowej opcji *Page Properties*.



Rysunek 1.8 Wprowadzanie nazwy strony w systemie DAQFactory

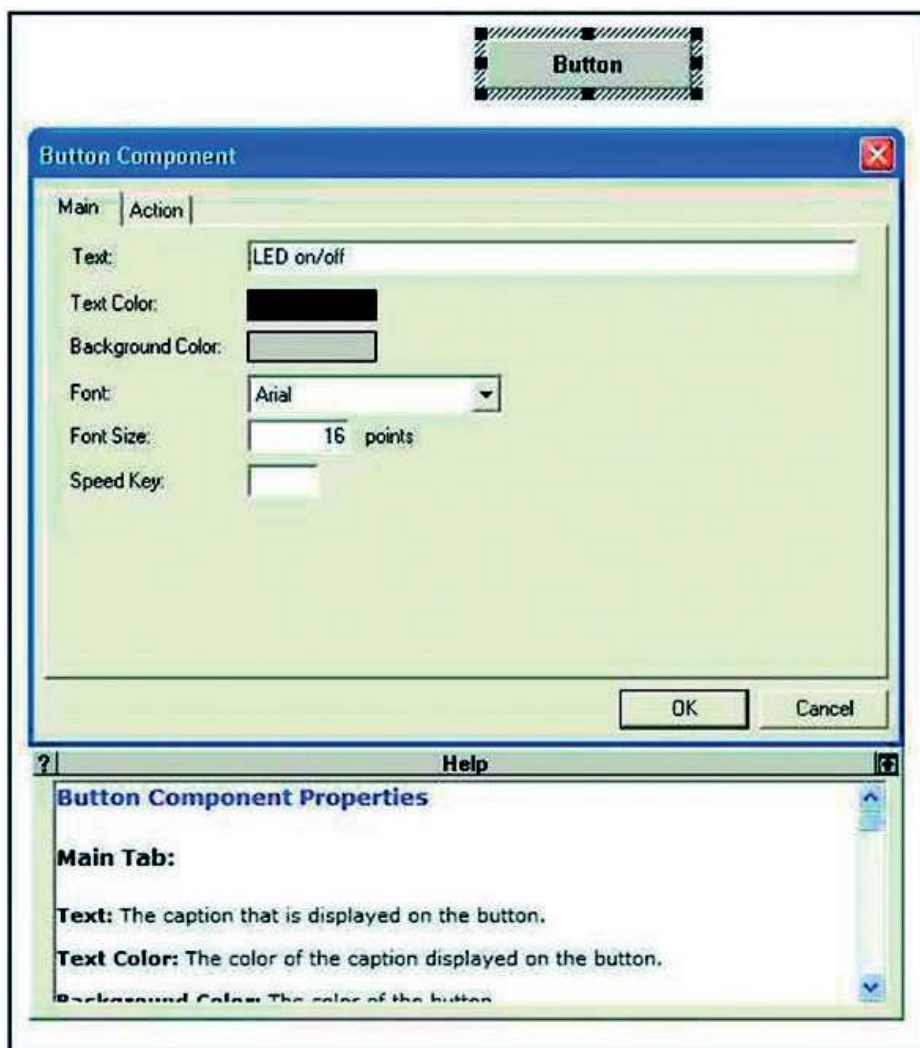
Kontrolka przycisku jest wybierana po kliknięciu prawym przyciskiem myszy opcji menu podręcznego. Jeśli zostanie wciśnięty klawisz *Ctrl*, przycisk może zostać umieszczony pod odpowiednim tekstem.

Po umieszczeniu komponentu przycisku na ekranie można go skonfigurować, wypełniając odpowiednie zakładki w oknie dialogowym właściwości. Okno to zostanie wyświetlone po kliknięciu kontrolki prawym przyciskiem myszy, a następnie wybraniu opcji *Properties* (właściwości), jak pokazano na rysunku 1.9. Aby połączyć przycisk na ekranie z akcją realizowaną w środowisku eksperymentalnym, należy wybrać opcję *Do Action* (wykonaj akcję).



Rysunek 1.9 Menu podręczne umożliwiające wyświetlenie właściwości przycisku

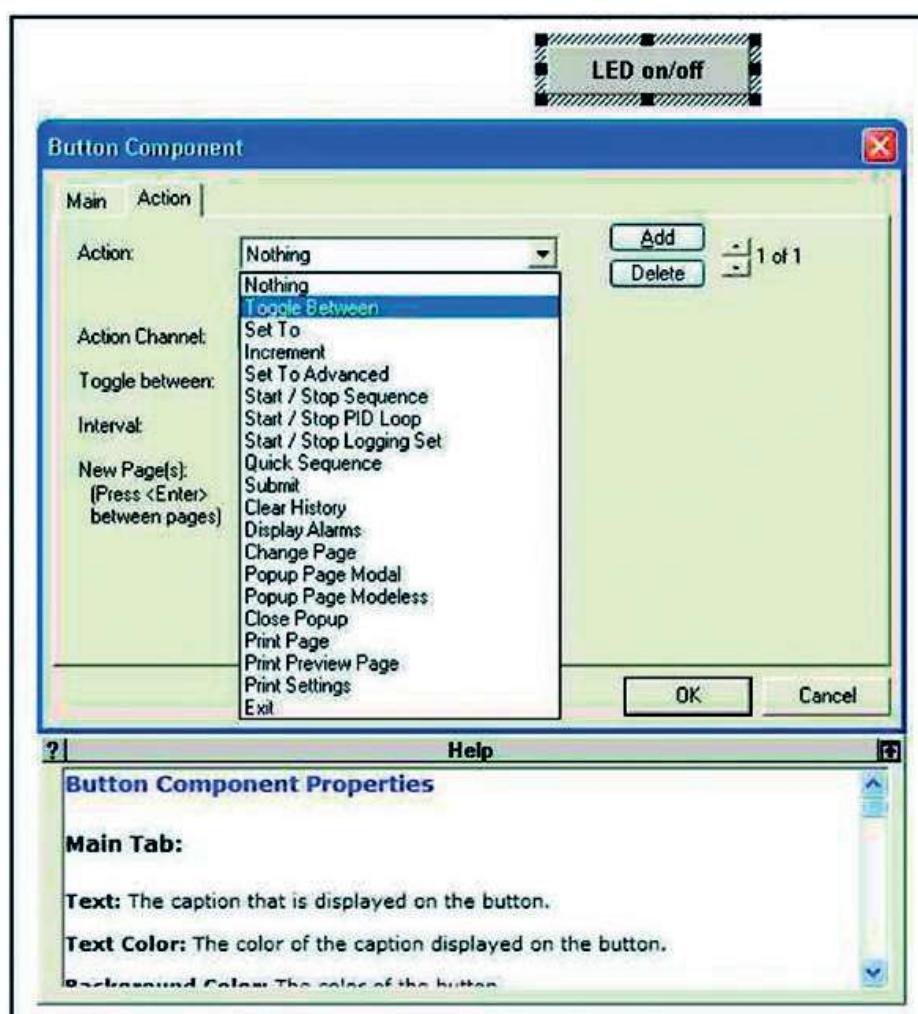
Jak widać na rysunku 1.10, okno właściwości przycisku zawiera dwie zakładki: *Main* (główne) i *Action* (akcja).



Rysunek 1.10 Zakładka Main w oknie dialogowym właściwości przycisku

Zakładka *Main* pozwala wybrać dla przycisku określoną czcionkę i jej rozmiar, a także odpowiedni kolor. Tekst na wyświetlanej kontrolce zostanie domyślnie wypośrodkowany. Jak widać na rysunkach 1.6 i 1.8, pod oknem właściwości jest wyświetlana pomoc kontekstowa, z której można skorzystać podczas konfigurowania komponentu ekranowego. Zawartość tekstową okna pomocy można przeglądać za pomocą kontrolki przewijania znajdujących się przy jego prawej krawędzi.

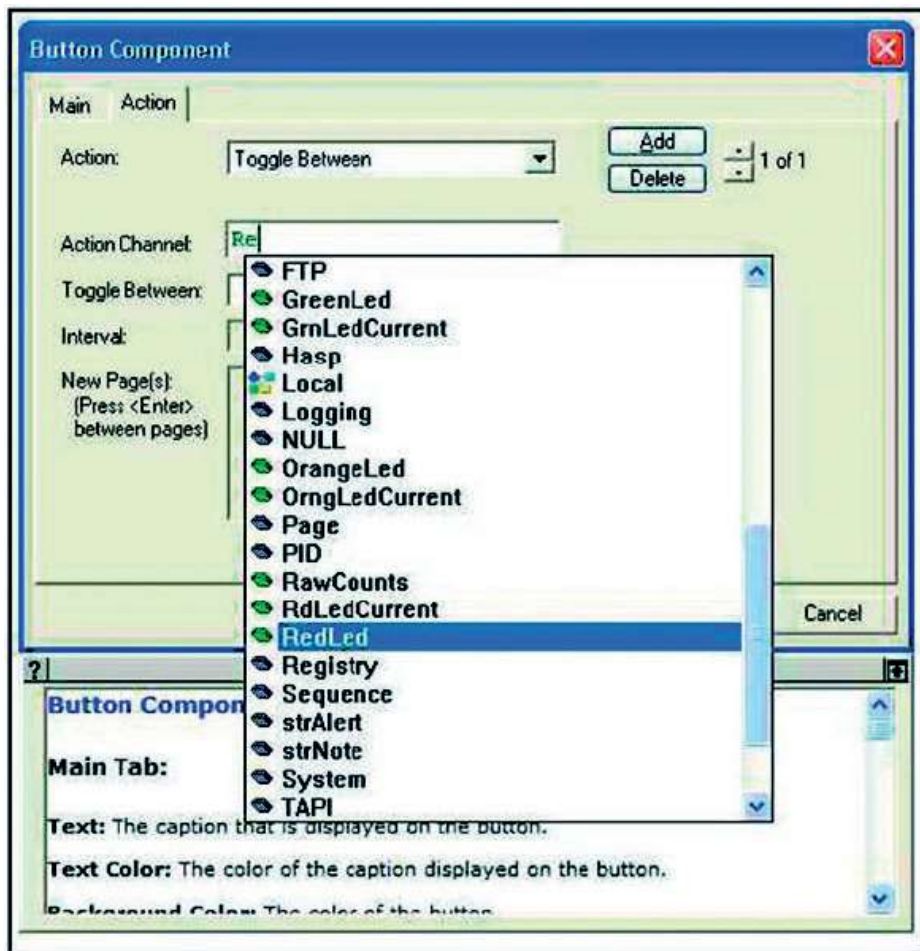
Druga zakładka (*Action*) została zaprezentowana na rysunku 1.11.



Rysunek 1.11 Zakładka Action z otwartą listą rozwijalną, pozwalającą na wybranie akcji

Jak sama nazwa wskazuje, zakładka *Action* pozwala na skonfigurowanie akcji wykonywanej po kliknięciu kontrolki przycisku. Jak widać na rysunku 1.11, w zakładce istnieje lista rozwijalna akcji lub opcji, z której można dokonać wyboru. Szczegóły dotyczące każdej z pozycji, dostępnej na liście, są wyświetlane w obszarze pomocy, widocznym w dolnej części okna.

W celu ręcznego sterowania diodą LED, zamontowaną na płytce prototypowej, należy wybrać z listy rozwijanej opcję *Toggle Between* (przełącz pomiędzy). Po jej wybraniu trzeba następnie wypełnić kilka kolejnych okien dialogowych, które dokładnie określają, co powinno zostać wykonane (rysunek 1.12).

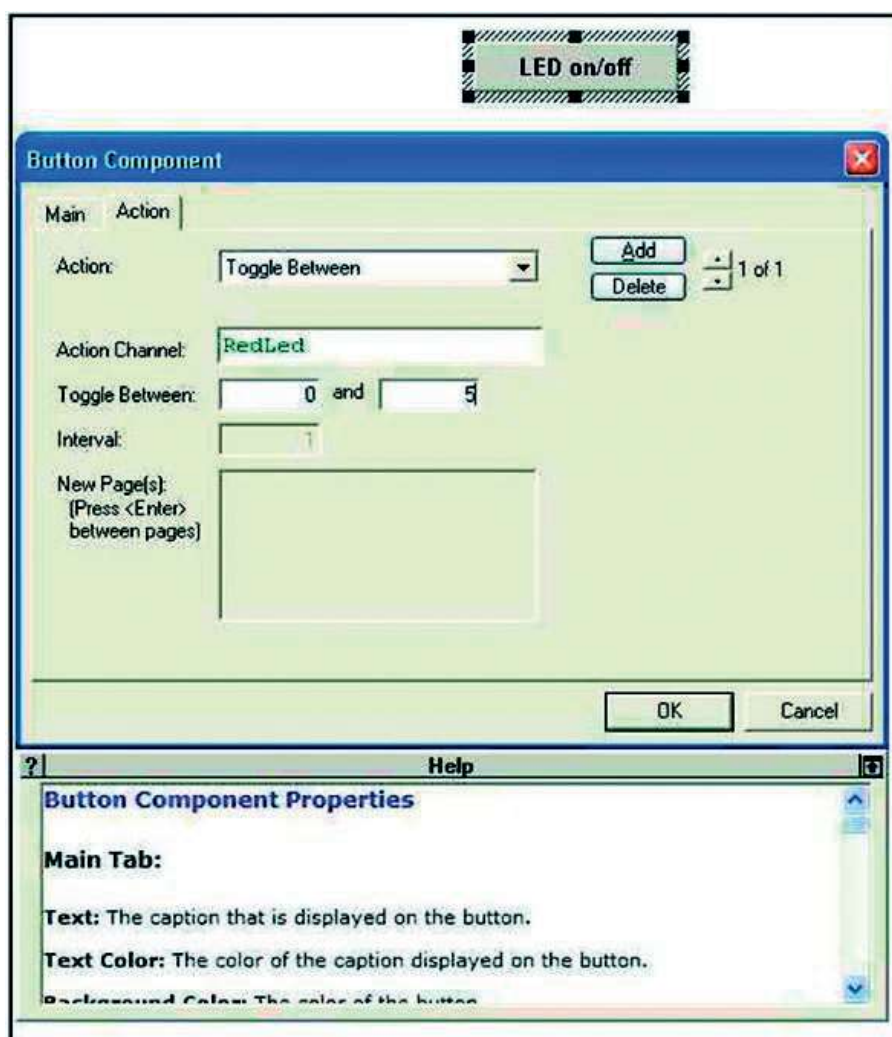


Rysunek 1.12 Wybór nazwy kanału w przypadku akcji *Toggle Between*

Opcja *Toggle Between* jest zazwyczaj związana z kanałem, który powinien być zasilany dwoma alternatywnymi napięciami, takimi jak 0 i 5 V. Jak napisano na początku tego rozdziału, wprowadzenie danych konfiguracyjnych kanału wejścia i wyjścia doprowadziło do powstania pozycji *RedLed* na inteligentnej liście wyboru, widocznej na rysunku 1.12. Dwukrotne kliknięcie wpisu spowoduje wprowadzenie wybranej nazwy kanału do pola edycji *Action Channel* (kanał akcji). Należy się upewnić, że nazwa jest poprawna i nie zostały przypadkowo dodane do niej (ani z niej usunięte) żadne znaki.

W zakładce *Action* widoczne są również niedostępne opcje, które zostają aktywowane po wybraniu akcji z listy. Jak widać na rysunku 1.1.3, w tym przypadku znajdują się tam pola do wprowadzania przełączanych wartości, które zostaną odwzorowane na osie wykresu graficznego.

Celem tego ćwiczenia jest bezpośrednie sprzężenie przycisku ze sterownikiem zasilania diody LED bez potrzeby tworzenia skryptu.



Rysunek 1.13 Wypełnienie wszystkich niezbędnych pól w zakładce Action przycisku

Przebieg eksperymentu, testy i proces projektowania

Dodatni przewód woltomierza cyfrowego należy podłączyć w urządzeniu LabJack do zacisku wejścia i wyjścia o numerze 0, a czarny przewód do zacisku GND. Skala miernika powinna zostać ustawiona tak, aby można było zmierzyć wartość 5 V. Następnie należy włączyć miernik i upewnić się, że pokazuje wartość zerową. Po kliknięciu na ekranie przycisku włączania/wyłączania diody LED miernik powinien pokazać 5 V.

Jeśli napięcie 5 V nie pojawia się, należy sprawdzić, czy we wszystkich komponentach i listach występuje poprawna nazwa kanału. Środowisko rozróżnia wielkość liter, a więc nazwa *RedLed* musi występować dokładnie w takiej formie we wszystkich komponentach ekranu i na liście kanałów. Oprócz tego należy się upewnić, że wartości *Toggle Between* wynoszą 0 i 5 V.

Po uzyskaniu sygnału 5 V na złączu LabJack można przygotować podłączenie do układu scalonego CD4050. W tym celu należy umieścić ten układ we właściwym miejscu płytki prototypowej, czyli wzdłuż centralnej szczeliny dzielącej. Zwyczajowo układ scalony umieszcza się tak, aby wyprowadzenie numer 1 znajdowało się w lewym dolnym narożniku (gdy patrzy się na niego z góry).

Następnie można już podłączyć końcówkę 1 do dodatniego bieguna zasilania na płytce prototypowej, końcówkę 8 do ujemnego, a przewód prowadzący z zacisku 0 do końcówki o numerze 3. Zacisk GND z urządzenia LabJack powinien zostać podłączony do ujemnego bieguna zasilania na płytce prototypowej. Po podłączeniu zewnętrznego zasilania do tej płytki powinno się podłączyć woltomierz w taki sposób, aby dodatni przewód dotykał końcówki 2 układu scalonego, a ujemny ujemnego bieguna linii zasilania. Kliknięcie na ekranie przycisku włączania/wyłączania diody LED powinno spowodować wyświetlenie przez miernik napięcia +5 V.

Jeśli sygnał 5 V nie pojawia się na mierniku, należy sprawdzić najpierw zasilanie, a następnie prześledzić wszystkie połączenia przewodowe, które pokazano na schemacie z rysunku 1.3.

Gdy na wyprowadzeniu 2 układu CD4050 pojawi się sygnał 5 V, który może być włączany/wyłączany za pomocą przycisku na ekranie, wtedy można będzie wykonać obliczenia związane z ograniczeniem prądu zasilającego czerwoną diodę LED. Aby wyznaczyć wartość oporności, wymaganą do ograniczenia prądu do połowy zakresu sugerowanego przez producenta, należy wykorzystać prawo Ohma. Z karty katalogowej wynika, że przez jasno świecącą diodę LED, użytą w projekcie autora, może przepływać maksymalny prąd 30 mA, który powoduje nominalny spadek napięcia 1,8 V. Zgodnie z prawem Ohma, rezystor o wartości 213 Ω ograniczy prąd diody do połowy dopuszczalnej wartości maksymalnej. Każdy standardowy rezystor o wartości 220 Ω lub większej mógłby więc zabezpieczyć tę diodę LED. W niniejszym układzie eksperymentalnym został użyty rezystor 470 Ω .

Dobrym zwyczajem jest obliczenie oporności, która pozwoli uzyskać maksymalną, dopuszczalną wartość natężenia prądu diody LED, określoną w karcie katalogowej przy uwzględnieniu nominalnego napięcia zasilania. W ten sposób można wyznaczyć wartość rezystora zabezpieczającego diodę. Jeśli zostanie wybrana następna wartość z szeregu standardowego, większa od wyliczonej minimalnej, dioda wciąż będzie dobrze świecić, a dodatkowo uzyska dodatkowy margines ochrony przed przeciążeniem, co wydłuży jej żywotność i pomoże ograniczyć prąd pobierany z zasilacza.