



SparkFun Inventor's Kit

WERSJA 4.1a

Twoja instrukcja do
SIK SparkFun RedBoard

SparkFun Inventor's Kit, wersja 4.1a

WITAMY W INSTRUKCJI DO SPARKFUN INVENTOR'S KIT (SIK).

To jest Twoja mapa do nawigacji po początkach elektroniki wbudowanej. Ta broszura zawiera wszystkie informacje potrzebne do zbudowania pięciu projektów obejmujących 16 obwodów SIK dla płytki SparkFun RedBoard.

W centrum tego podręcznika znajduje się jedna podstawowa filozofia: że każdy może (i powinien) bawić się elektroniką.

Kiedy skończysz z tym przewodnikiem, zbudujesz pięć świetnych projektów i zdobędziesz know-how do tworzenia niezliczonych innych.

A teraz dość gadania - zacznijmy coś robić! Aby uzyskać cyfrową wersję tego przewodnika z bardziej szczegółowymi informacjami na temat i linki wyjaśniające istotne terminy i pojęcia, odwiedź stronę:

sparkfun.com/SIKguide

Polskie tłumaczenie:

Szymon Reiter, Fundacja Hackerspace Kraków



Zawartość

2 INTRO

- 2** Platforma RedBoard
- 3** Montaż płyty bazowej
- 4** Anatomia RedBoard
- 5** Anatomia płytka prototypowej
- 6** Arduino IDE
- 10** Spis części

12 PROJEKT 1: ŚWIATŁO

- 13** Obwód 1A: Miganie diody LED
- 20** Obwód 1B: Potencjometr
- 26** Obwód 1C: Fotorezystor
- 31** Obwód 1D: Lampka nocna RGB

36 PROJEKT 2: DŹWIĘK

- 37** Obwód 2A: Brzęczyk
- 42** Obwód 2B: Cyfrowa trąbka
- 47** Obwód 2C: Gra „Simon mówi”

53 PROJEKT 3: RUCH

- 54** Obwód 3A: Serwomechanizmy
- 60** Obwód 3B: Czujnik odległości
- 65** Obwód 3C: Wykrywanie ruchu

71 PROJEKT 4: WYSWIETLACZ

- 72** Obwód 4A: LCD „Hello, World!”
- 77** Obwód 4B: Czujnik temperatury
- 82** Obwód 4C: Gra „Kim jestem?”

88 PROJEKT 5: ROBOT

- 89** Obwód 5A: Silnik elektryczny
- 96** Obwód 5B: Zdalnie sterowany robot
- 102** Obwód 5C: Robot autonomiczny

106 CO DALEJ?

Platforma RedBoard

REWOLUCJA DLA MAJSTERKOWICZÓW: W SparkFun wierzymy, że zrozumienie elektroniki jest podstawową umiejętnością, która otwiera świat możliwości w dziedzinie robotyki, Internetu rzeczy (IoT), inżynierii, modzie, przemyśle medycznym, naukach o środowisku, sztuk performatywnych i nie tylko. Niniejszy przewodnik ma na celu pokazanie połączenia między oprogramowaniem i sprzętem, wprowadzając w Arduino i części SparkFun, ponieważ przydają się do tworzenia ciekawych projektów. Obwody w tym przewodniku rozwijają się w miarę wprowadzania nowych koncepcji i komponentów. Ukończenie każdego obwodu oznacza znacznie więcej niż tylko „eksperymentowanie”, odejdziesz z zabawnym projektem, którego możesz użyć i poczuciem spełnienia, które jest dopiero początkiem przygody z elektroniką.

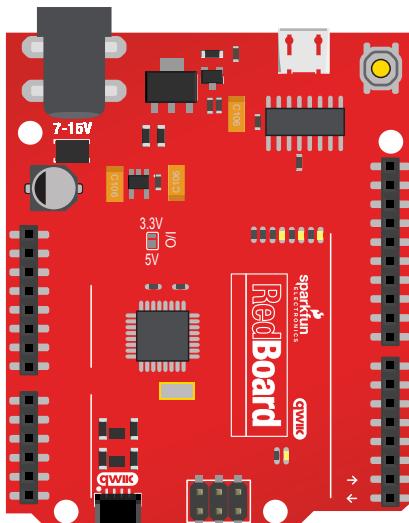
Na końcu każdego obwodu znajdziesz wyzwania związane z kodowaniem, które poszerzą twoją wiedzę i napędzają ciągle innowacje.

KOMPUTERY I ŚWIAT FIZYCZNY

SparkFun RedBoard Qwiic jest platformą programistyczną. U swoich podstawa RedBoard jest zasadniczo małym, przenośnym komputerkiem. Jest on zdolny do przyjmowania danych wejściowych (takich jak naciśnięcie przycisku lub odczyt z czujnika światła) i interpretowania tych informacji do sterowania różnymi wyjściami (takimi jak miganie LED lub obracanie silnikiem elektrycznym).

Płytkę ta jest w stanie przenieść świat elektroniki i powiązać go światem fizycznym w rzeczywisty i namalowany sposób.

SPARKFUN REDBOARD QWIIC jest jedną z wielu płyt rozwojowych opartych na mikroprocesorze ATmega328. Posiada 14 cyfrowych pinów wejściowych/wyjściowych (z których sześć może być wyjściami PWM), sześć wejść analogowych, oscylator kwarcowy 16 MHz, złącze USB, gniazdo zasilania i przycisk resetowania, gniazdo zasilania i przycisk resetowania. Dowiesz się więcej o każdej z funkcji RedBoard w trakcie wycieczki po tym przewodniku



NOTE: For the remainder of this guide, in the interest of brevity, we will refer to the RedBoard Qwiic simply as the “RedBoard.”

Montaż płyty bazowej

Przed przystąpieniem do tworzenia obwodów, należy najpierw zmontować płytę bazową. To urządzenie ułatwia budowanie obwodów, utrzymując mikrokontroler RedBoard i płytę stykową (breadboard) bez obawy o odłączenie lub uszkodzenie obwodu.



ABY ROZPOCZĄĆ, zbierz części: płytę RedBoard, płytę prototypową, dołączony śrubokręt, płytę bazową i dwie śruby płyty bazowej.

Śrubokręt ma zarówno płaski jak i krzyżakowy (Philips) koniec. Wystarczy obrócić jego część roboczą.



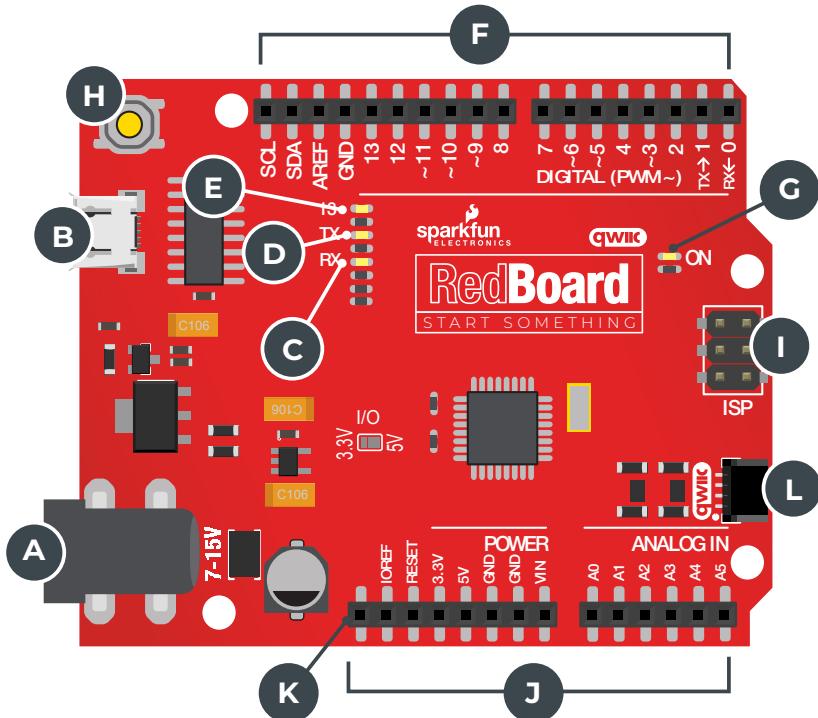
ODKLEJ papier zabezpieczający taśmę samoprzylepną z płytki prototypowej.

OSTROŻNIE WYRÓWNAJ płytę nad jej miejscem na płytcie bazowej. Tekst na płytce powinien być skierowany w tym samym kierunku, co tekst na płytcie bazowej. Mocno dociśnij płytę do płyty bazowej, aby ją przykleić.



WYRÓWNAJ RedBoard w jej miejscu na płytcie bazowej. Tekst na niej powinien być skierowany w tym samym kierunku, co tekst na płytcie prototypowej i płytcie bazowej. Używając jednej z dwóch dołączonych śrub, przymocuj płytę RedBoard do jednego z czterech otworów dystansowych znajdujących się na płytcie bazowej. Plastikowe otwory nie są gwintowane, więc należy zastosować nacisk podczas przekręcania śrubokręta. Wkręć drugą śrubę w otwór po przekątnej w stosunku do pierwszej. W ten sposób płyta bazowa jest teraz trwale zmontowana.

Anatomia RedBoarda



ELEMENTY REDBOARDA

A	GNIAZDO ZASILANIA	Może być użyte z zasilaczem wtyczkowym 9-12V
B	ZASILANIE PORT USB	Zapewnia zasilanie i komunikację z komputerem za pośrednictwem portu USB
C	LED (RX: ODBIÓR)	Pokazuje kiedy płytka odbiera dane za pośrednictwem USB
D	LED (TX: NADAWANIE)	Pokazuje kiedy płytka nadaje dane za pośrednictwem USB
E	LED PIN D13	Ta dioda jest podłączona stałe do pinu numer 13 Możesz ją użyć w swoim oprogramowaniu
F	PINY GROUND, DIGITAL, RX, TX, SDA, SCL	Te piny mogą być użyte jako wejścia, wyjścia albo zasilanie
G	POWER LED	Świeci, gdy płytka jest zasilana
H	PRZYCISK RESET	Ręczny reset, powoduje wykonanie Twojego programu od początku
I	ZŁĄCZE ISP	To złącze może być użyte do awaryjnego zaprogramowania mikrokontrolera znajdującego się na płytce
J	WEJŚCIA ANALOGOWE ZASILANIA, MASY	Pin zasilania i masy dostarczają zasilanie do układów peryferyjnych, wejścia analogowe potrafią zmierzyć podane na nie napięcie
K	RFU	Pin zarezerwowany do przyszłego użycia
L	ZŁĄCZE QWIIC®	Złącze Qwiic® służy do komunikacji za pomocą magistrali I ² C

Anatomia breadboardu

Płytką prototypowa to platforma do tworzenia obwodów, która umożliwia łączenie wielu komponentów bez użycia lutownicy.

MAGISTRALA ZASILANIA

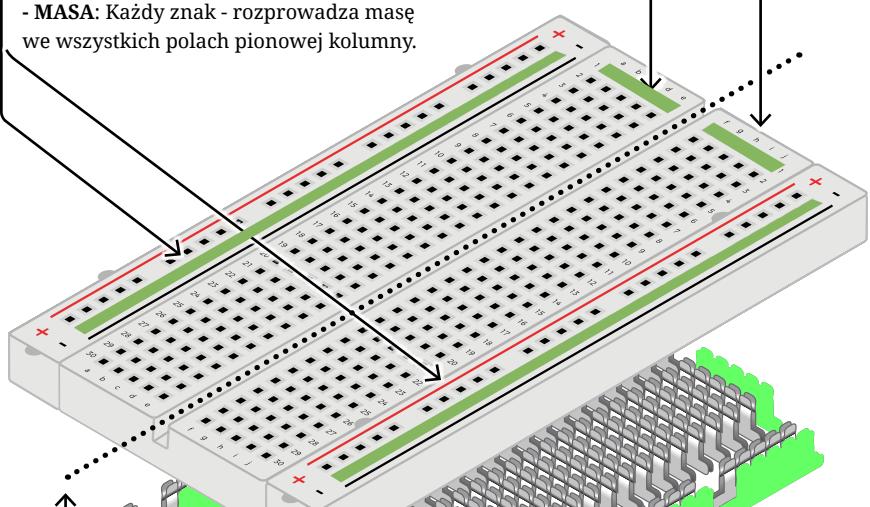
Każda strona płytka stykowej ma parę pionowych połączeń oznaczonych - i +

+ ZASILANIE: Każdy znak + rozprowadza zasilanie we wszystkich polach pionowej kolumny.

- MASA: Każdy znak - rozprowadzi masę we wszystkich polach pionowej kolumny.

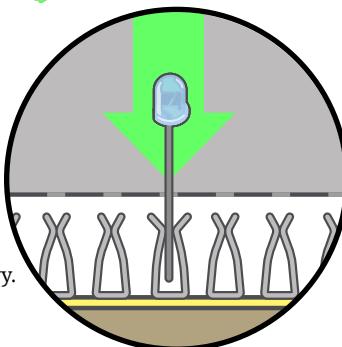
POZIOME RZĘDY

Każde z 5 pól oznaczonych a-e oraz f-j są ze sobą połączone. Komponenty podłączone do rzędu będą połączone z każdym innym elementem w tym samym rzędzie.



LINIA ŚRODKOWA

Ta linia dzieli płytkę prototypową na pół, ograniczając elektryczność do jednej lub drugiej połowy.



WYKONYWANIE

POŁĄCZEŃ

Większość komponentów w tym zestawie pasuje do breadboardów i może być łatwo podłączona oraz usunięta.

Arduino IDE

ABY URUCHOMIĆ REDBOARD.

należy pobrać najnowszą wersję oprogramowania Arduino ze strony www.arduino.cc (za darmo!).

Oprogramowanie to, znane jako Arduino IDE (Integrated Development Environment), pozwoli ci zaprogramować RedBoard tak, aby robił dokładnie to, co chcesz.

To jak edytor tekstu do programowania.

Korzystając z komputera z dostępem do Internetu, otwórz ulubioną przeglądarkę i wpisz następujący adres URL w pasku adresu:

TUTAJ POBIERZ ARDUINO: arduino.cc/downloads



1. POBIERZ I ZAINSTALUJ ARDUINO IDE

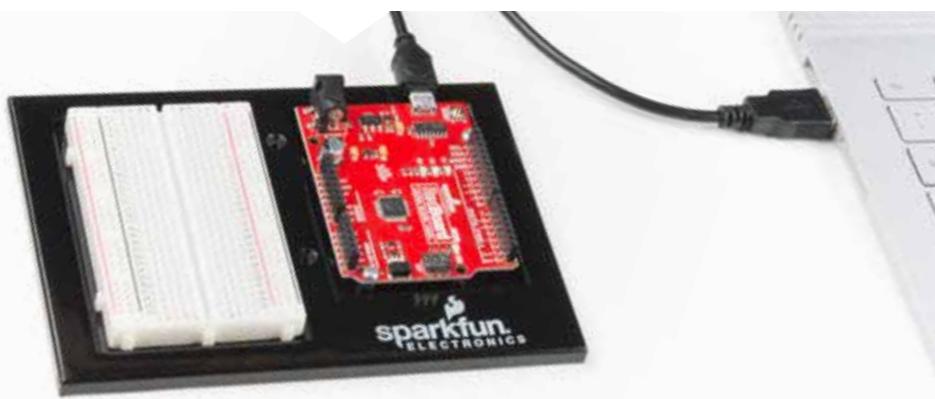
Wybierz opcję instalatora odpowiednią dla używanego systemu operacyjnego. Po zakończeniu pobierania otwórz plik i postępuj zgodnie z instrukcjami instalacji.

2. INSTALACJA STEROWNIKÓW USB

Aby sprzęt RedBoard działał z systemem operacyjnym komputera, należy zainstalować kilka sterowników. Przejdz do www.sparkfun.com/ch340, aby uzyskać szczegółowe instrukcje dotyczące instalowania sterowników USB na komputerze.

3. PODŁĄCZ REDBOARD DO KOMPUTERA

Użyj kabla USB dostarczonego w SIK, aby podłączyć RedBoard do jednego z wejść USB komputera.



4. POBIERZ I ZAINSTALUJ SIK CODE

Każdy z obwodów, które zbudujesz w zestawie SparkFun Inventor's Kit, ma już napisany szkic kodu Arduino. Ten przewodnik pokaże ci, jak się w tym poruszać aby sterować sprzętem.

POBIERZ KOD TUTAJ:

sparkfun.com/SIKcode

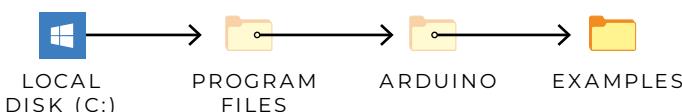
SKOPIUJ "SIK GUIDE CODE" DO KATALOGU "EXAMPLES" W KATALOGU ARDUINO

Przeglądarka automatycznie pobierze plik lub zapyta, czy chcesz pobrać plik .zip. Wybierz opcję „Zapisz plik”. Zlokalizuj kod (zwykle w katalogu przeglądarki „Pobrane”). Będziesz musiał przenieść go do podkatalogu „Examples” w swojej instalacji Arduino IDE aby działał prawidłowo.

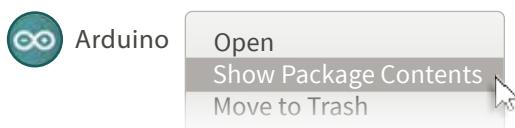
Rozpakuj plik „SIK GUIDE CODE”.

Powinieneś on znajdować się w katalogu „Pobrane” przeglądarki. Kliknij prawym przyciskiem myszy (lub ctrl + kliknięcie) spakowany folder i wybierz opcję „rozpakuj”.

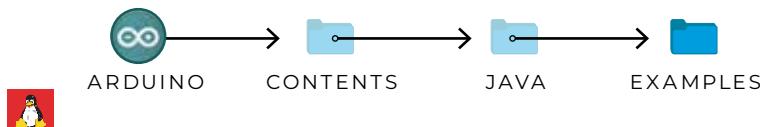
WINDOWS: Skopiuj lub przenieś rozpakowane pliki „SIK Guide Code” z katalogu „Downloads” do katalogu „Examples” aplikacji Arduino.



MAC OS: Znajdź „Arduino” w katalogu „Aplikacje” w Finderze. Kliknij prawym przyciskiem myszy (ctrl + kliknięcie) na „Arduino” i wybierz „Pokaż zawartość pakietu”.



Skopiuj lub przenieś rozpakowany katalog „SIK Guide Code” z katalogu „Downloads” do katalogu aplikacji Arduino o nazwie „Examples”.

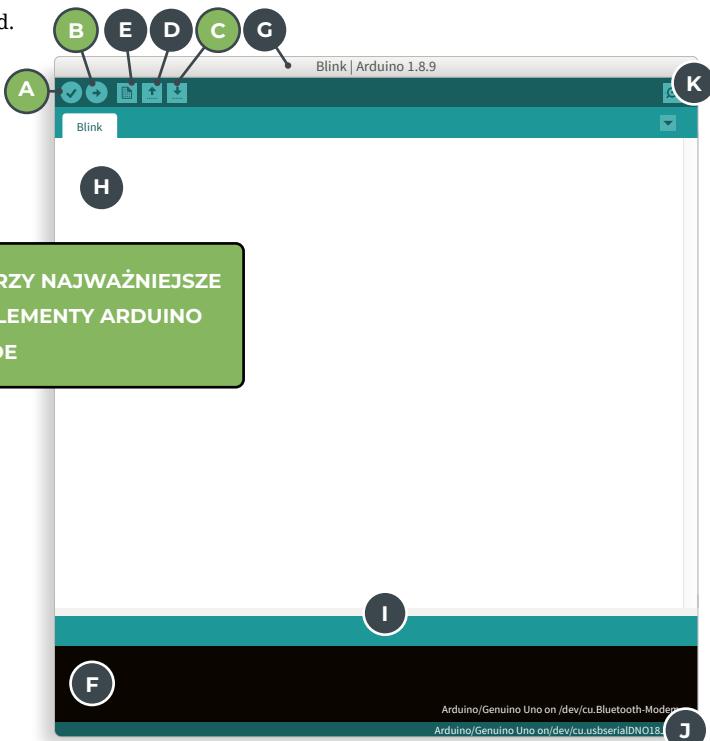


LINUX: Instrukcje specyficzne dla Twojej dystrybucji znajdziesz pod adresem:

www.sparkfun.com/ch340

5. OTWÓRZ ARDUINO IDE

Otwórz oprogramowanie Arduino IDE na swoim komputerze. Poszperaj i poznaj interfejs. Nie będziemy od razu kodować, ten krok polega na ustawieniu Twojego IDE komunikacji z płytą RedBoard.



GRAFICZNY INTERFEJS UŻYTKOWNIKA (GUI)

A	WERYFIKUJ	Kompiluje i zatwierdza Twój kod. Wychwyci błędy w składni (takie jak brakujące średniki lub nawiasy).
B	WGRAJ	Wysyła skompilowany kod do płytki RedBoard
C	ZAPISZ	Zapisuje aktywny szkic.
D	OTWÓRZ	Otwiera zapisany wcześniej szkic.
E	NOWY	Otwiera nowe okno z pustym kodem.
F	OKNO DEBUGOWANIA	Wyświetla wszelkie błędy wyłapane podczas komplikacji.
G	NAZWA SZKICU	Pokazuje nazwę szkicu.
H	OBSZAR NA KOD	Miejsce, w którym tworzysz lub edytujesz kod swojego szkicu.
I	OBSZAR KOMUNIKATÓW	Wskazuje, czy kod się kompluluje, przesyła lub zawiera błędy.
J	OBSZAR KOMUNIKACJI	Wyświetla aktualnie wybraną płytę oraz port szeregowy
K	MONITOR PORTU SZEREGOWEGO	Otwiera okno wyświetlające wszelkie informacje z portu szeregowego przesypane przez RedBoard (przydatne do debugowania).

6. WYBIERZ SWOJĄ PŁYTKĘ I PORT SZEREGOWY

NOTATKA: Płytkę SparkFun RedBoard oraz Arduino/Genuino UNO można stosować wymiennie, ale nie znajdziesz RedBoard na liście w Arduino. Zamiast tego wybierz ARDUINO/GENUINO UNO.

The diagram illustrates the process of selecting a board and a serial port in the Arduino IDE across three operating systems: Windows, Mac OS, and Linux.

WYBIERZ SWOJĄ PŁYTKĘ (Windows): Tools > Board > Arduino/Genuino Uno

WYBIERZ PORT SZEREGOWY (WINDOWS): Tools > Port > COM#XX

WYBIERZ SWÓJ PORT (Linux): Tools > Port > /dev/cu.usbserialXXXXXXX

WYBIERZ SWÓJ PORT (Mac OS): Tools > Port > /dev/cu.usbserialDNO18JWS

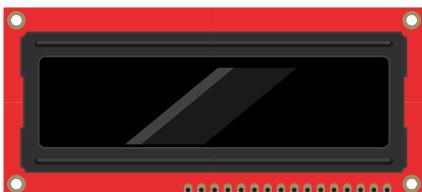
NOTA: The text "WIELKOŚĆ LITER MA ZNACZENIE!" is located at the bottom right of the Linux section.

NOTA: A link to "http://arduino.cc/playground/Learning/Linux" is located at the bottom left of the Linux section.

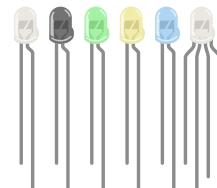
Spis części

Zestaw SparkFun Inventor's Kit zawiera szeroką gamę komponentów elektronicznych. Jak przejdziesz przez każdy obwód, nauczysz się korzystać z nowych i bardziej skomplikowanych części do wykonywania coraz bardziej złożonych zadań.

WYSWIETLACZ LCD



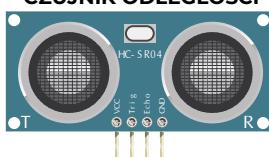
DIODY ŚWIĘCĄCE



POTENCJOMETR



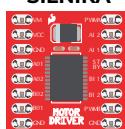
ULTRADŹWIĘKOWY CZUJNIK ODLEGŁOŚCI



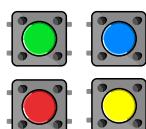
BUZZERETR



STEROWNIK SILNIKA



PRZYCISK



REZYSTOR 10kΩ



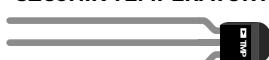
REZYSTOR 330Ω



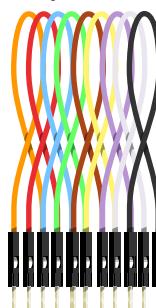
FOTOREZYSTOR



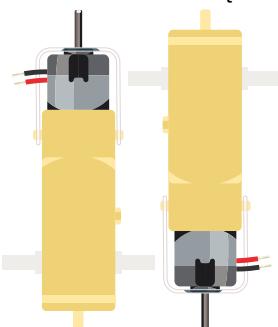
CZUJNIK TEMPERATURY



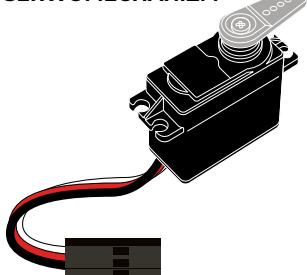
KABELKI POŁĄCZENIOWE



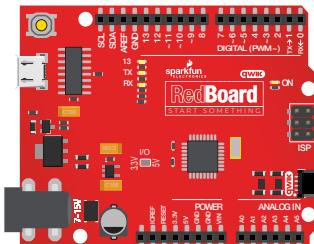
SILNIKI Z PRZEKŁADNIĄ



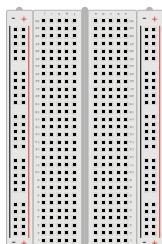
SERWOMECHANIZM

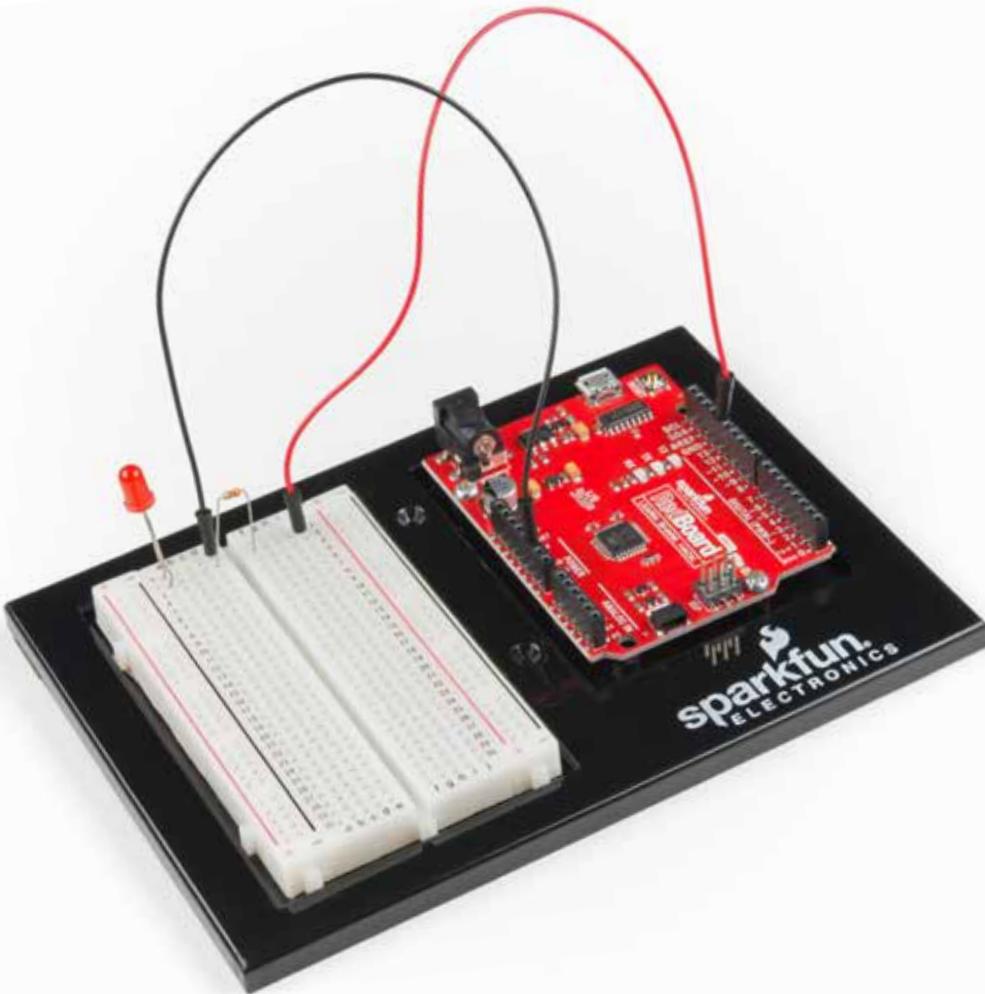


SPARKFUN REDBOARD

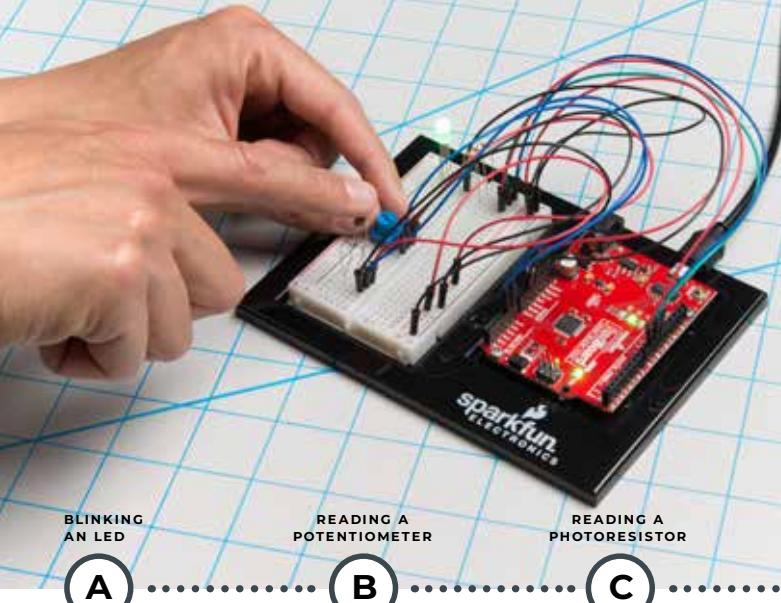


BREADBOARD





Zacznijmy
od Twojego
pierwszego
OBWODU!



BLINKING
AN LED

READING A
POTENTIOMETER

READING A
PHOTORESISTOR

RGB NIGHT-LIGHT

A

B

C

D

PROJEKT 1

Witamy w pierwszym projekcie zestawu SparkFun Inventor's Kit. Każdy **projekt** jest podzielony na kilka **obwodów**, z których ostatni jest zwieńczeniem technologii, które pojawiły się wcześniej. Łącznie dostępnych jest pięć projektów, z których każdy ma pomóc Ci poznać nowe techniki i koncepcje. Ten pierwszy projekt stworzy podwaliny pod resztę i pomoże Ci zrozumieć podstawy budowy obwodów i elektryczności!

W Projekcie 1 dowiesz się o **diodach elektroluminesencyjnych (LED)**, **rezystorach**, **wejściach, wyjściach i czujnikach**. Pierwszym projektem będzie zbudowanie i zaprogramowanie własnej wielokolorowej lampki nocnej! Lampka nocna wykorzystuje czujnik do włączania diody LED RGB (czerwonej, zielonej, niebieskiej), gdy robi się ciemno, a kolor można zmienić za pomocą pokrętła wejściowego.

NOWE IDEE

Każdy projekt będzie wprowadzał nowe koncepcje i komponenty, które zostaną opisane bardziej szczegółowo w miarę postępów w kolejnych częściach.

NOWE KOMPONENTY W TYM PROJEKCIE

- DIODY ŚWIĘCĄCE
- REZYSTORY
- POTENCJOMETRY
- FOTOREZYSTORY

NOWE POJĘCIA W TYM PROJEKCIE

- POLARYZACJA
- PRAWO OHMA
- WYJŚCIA CYFROWE
- ANALOGOWY KONTRA CYFROWY
- WEJŚCIA ANALOGOWE
- KONwersja ANALOG-CYFRA
- DZIELNIK NAPIĘCIA
- MODULACJA SZEROKOŚCI IMPULSU
- FUNKCJE

CZEGO SIĘ NAUCZYSZ

- JAK PRZESŁAĆ PROGRAM DO PŁYTKI REDBOARD
- PODSTAWY BUDOWY OBWODÓW
- JAK KONTROLOWAĆ DIODY ŚWIĘCĄCE ZA POMOCĄ WYJŚĆ CYFROWYCH
- JAK ODCZYTYWAĆ DANE Z SENSORÓW ZA POMOCĄ WEJŚĆ ANALOGOWYCH

Projekt 1A

Miganie diodą

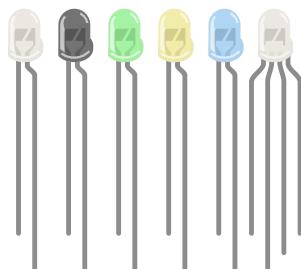
Diody LED można znaleźć w niemal każdym źródle światła, od żarówek oświetlających dom po małe lampki stanu migające na domowych urządzeniach elektronicznych. Miganie diody LED to klasyczny punkt wyjścia do nauki programowania wbudowanej elektroniki. To „Witaj, świecie!” mikrokontrolerów. Do tego obwodu napiszesz kod, który będzie włączał i wyłączał diodę LED.



YOU
NEED

NOWE ELEMENTY

Diody elektroluminescencyjne (LED) to małe lampki wykonane z diody krzemowej. Występują w różnych kolorach, jasnościach i rozmiarach. Diody LED mają **dodatnią (+)** i **ujemną (-)** nóżkę i przepuszczają prąd tylko w jednym kierunku. Diody LED mogą się również przepalić, jeśli przepływnie przez nie zbyt dużo prądu, dlatego zawsze należy używać rezystora, aby ograniczyć prąd podczas podłączania diody LED do obwodu.



REZYSTORY stawiają opór przepływowi prądu. Można ich używać do ochrony wrażliwych elementów takich jak diody LED. Siła rezystora (mierzona w omach) jest zaznaczona na korpusie rezystora za pomocą małych kolorowych pasków. Każdy kolor oznacza liczbę, którą można sprawdzić za pomocą tabeli rezystorów. Tabelę możecie znaleźć na końcu tej książki.

NOWE POJĘCIA

POLARYZACJA: Wiele elementów elektronicznych ma polaryzację, co oznacza, że prąd może (i powinien) przepływać przez nie tylko w jednym kierunku. Elementy spolaryzowane, takie jak dioda LED, mają dodatnią i ujemną końcówkę i działają tylko wtedy, gdy przepływa przez nie prąd w jednym kierunku. Niektóre elementy, takie jak rezystory, nie mają polaryzacji; prąd może przepływać przez nie w obu kierunkach.



PRAWO OHMA opisuje związek pomiędzy trzema podstawowymi elementami elektryczności: napięciem, rezystancją i prądem. Zależność tę można przedstawić za pomocą równania:

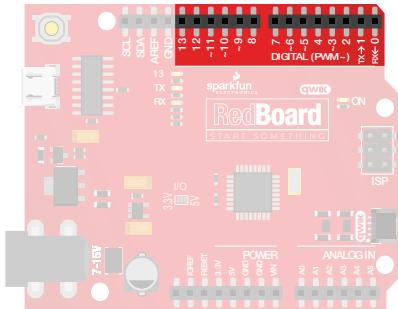
$$V=I \cdot R$$

V = napięcie w woltagach

I = prąd w amperach

R = rezystancja w omach (Ω)

Równanie to służy do obliczenia, jakie wartości rezystorów są odpowiednie, aby dostatecznie ograniczyć prąd płynący do diody LED, aby nie nagrzała się ona zbytnio i nie przepaliła.



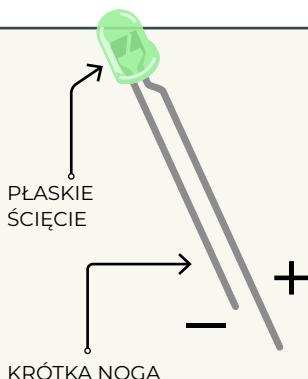
WYJŚCIE CYFROWE: Podczas pracy z płytami, takimi jak RedBoard, istnieje wiele pinów, do których można podłączyć komponenty elektroniczne. Podczas budowania układu ważna jest wiedza, które piny pełnią jakie funkcje. W tym układzie będziemy używać tak zwanego wyjścia cyfrowego. Na płycie RedBoard znajduje się 14 takich wyjść. Wyjście cyfrowe ma tylko dwa stany: **WŁĄCZONY** lub **WYŁĄCZONY**.

Te dwa stany można również określić jako **WYSOKI** lub **NISKI**, **PRAWDA** lub **FAŁSZ**. Kiedy dioda LED jest podłączona do jednego z tych pinów, pin może wykonywać tylko dwa zadania: włączać diodę LED i wyłączać diodę LED. Pozostałe piny i ich funkcje zbadamy w późniejszych układach.

NOWE IDEE

BEZPIECZEŃSTWO ELEKTRYCZNE: Nigdy nie pracuj nad obwodami, gdy płyta jest podłączona do źródła zasilania. SparkFun RedBoard działa przy napięciu 5 woltów, które choć nie jest wystarczające, aby spowodować obrażenia, wystarczy, aby uszkodzić elementy obwodu.

ORIENTACJA I POLARYZACJA KOMPONENTÓW: Instrukcje dotyczące orientacji każdego z nowych komponentów zostaną podane przed każdym schematem obwodu. Wiele komponentów ma polaryzację i tylko jedną prawidłową orientację, podczas gdy inne są niepolaryzowane.



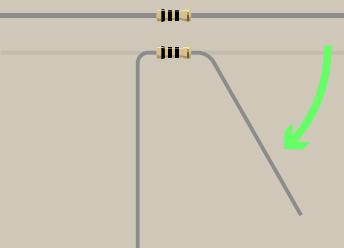
ELEMENTY POLARYZOWANE

Zwróć szczególną uwagę na diodę LED.

Ujemna strona diody LED to krótka nóżka, oznaczona płaską krawędzią.

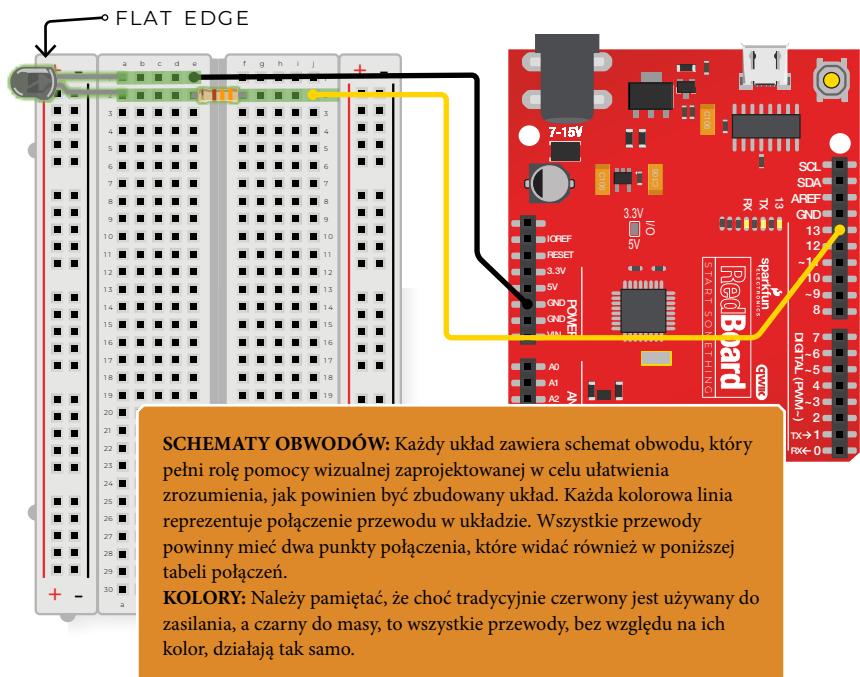
WYPROWADZENIA REZYSTORA

Elementy takie jak rezystory muszą mieć nogi wygięte pod kątem 90°, aby prawidłowo pasować do gniazd płytki prototypowej.



JAK TO POŁĄCZYĆ?

Sprawdź schemat obwodu i tabelę połączeń poniżej, aby zobaczyć, jak wszystko jest podłączone.



TABELE PODŁĄCZEŃ: Dla wielu początkujących elektroników posiadanie układu współrzędnych podczas budowania obwodów jest przydatne. Dla każdego układu znajdziesz tabelę połączeń, która zawiera współrzędne każdego komponentu lub przewodu oraz miejsce ich połączenia z płytą RedBoard, płytą stykową lub obydwoema. Płyta prototypowa ma układ współrzędnych liter/cyfr, podobnie jak gra okręty.

◆ D13 to ■ J2

...oznacza, że jeden koniec komponentu łączy się z cyfrowym pinem 13 na RedBoard, a drugi łączy się z J2 na płytce prototypowej

CONNECTION TYPES ◆ POŁĄCZENIA NA REDBOARD ■ POŁĄCZENIA NA BREADBOARDZIE

JUMPER WIRES

◆ D13 to ■ J2

◆ GND to ■ E1

LED

■ A1(-) to ■ A2(+)

330Ω RESISTOR
(ORANGE, ORANGE,
BROWN)

■ E2 to ■ F2

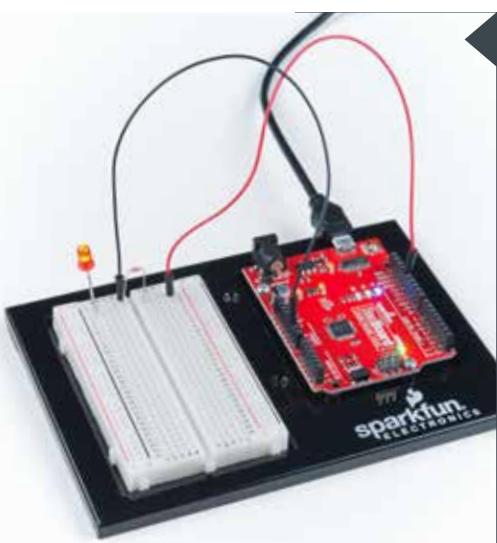
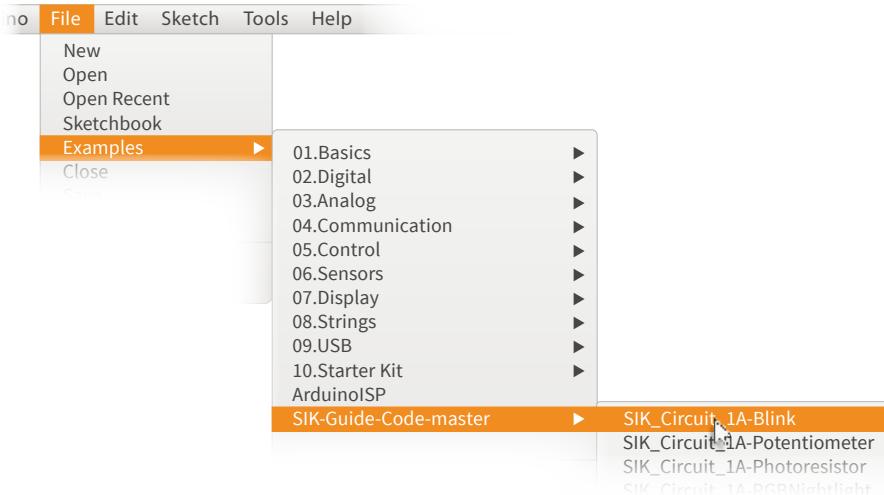
In this table, a yellow highlight indicates that a component has polarity and will only function if properly oriented.

Uruchom Arduino IDE

Podłącz płytke RedBoard do portu USB Twojego komputera

↑ Otwórz szkic: File > Examples > SIK-Guide-Code-master > CIRCUIT_1A-BLINK

→ Kliknij UPLOAD aby skompilować i wgrac program na płytke

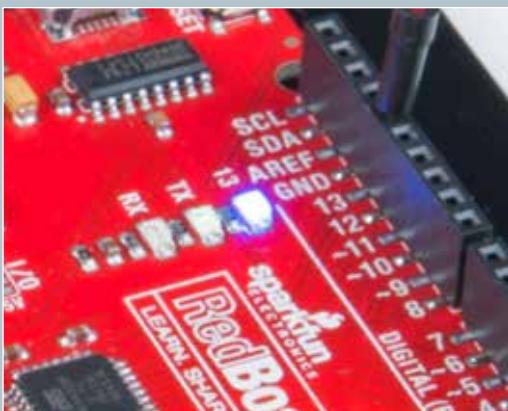


CZEGO SIĘ SPODZIEWAĆ?

Dioda LED będzie świecić przez dwie sekundy, a następnie zgaśnie na dwie sekundy. Jeśli tak nie jest, upewnij się, że poprawnie zmontowałeś obwód, zweryfikowałeś i przesłałeś kod do swojej płytki. Jeśli to nie zadziała, zobacz sekcję **Rozwiązywanie problemów** na końcu. Jednym z najlepszych sposobów zrozumienia przesłanego kodu jest zmiana czegoś i sprawdzenie, jak wpływa to na zachowanie obwodu. Co się stanie, jeśli zmienisz numer **delay (2000);** w jednej lub obu liniach kodu?

CO ROBI PROGRAM?

- 1 Włącza LED poprzez ustawienie stanu wysokiego (+5V) na cyfrowym wyjściu numer 13
- 2 Czeka 2 sekundy (2000 milisekund)
- 3 Wyłącza LED poprzez ustawienie stanu niskiego (0V) na wyjściu numer 13
- 4 Czeka 2 sekundy
- 5 Powtarza od początku.



ONBOARD LED PIN 13:

Być może zauważyłeś, że druga, mniejsza dioda LED migra jednocześnie z diodą LED w obwodzie płytki prototypowej. Nazywa się to wbudowaną diodą LED i można ją znaleźć na prawie każdej płytcie Arduino lub płytcie kompatybilnej z Arduino. W większości przypadków ta dioda LED jest podłączona do **cyfrowego wyjścia 13 (D13)**, tego samego styku używanego w tym obwodzie.

WAŻNE!

KOD: Szkice dołączone do każdego obwodu przedstawiają nowe techniki i koncepcje programowania w miarę postępów w przewodniku. Sekcja KOD podkreśla nowe i ważne linie kodu ze szkicu i wyjaśnia je bardziej szczegółowo.

KOD

SETUP I LOOP

`void setup(){} &`
`void loop(){}`

Każdy program Arduino potrzebuje tych dwóch funkcji. Kod umieszczony pomiędzy nawiasami klamrowymi {} funkcji **setup()** jest uruchamiany raz.

Kod znajdujący się pomiędzy nawiasami klamrowymi **loop()** {} jest wykonywany w kółko, dopóki RedBoard nie zostanie zresetowany lub wyłączony.

INPUT czy OUTPUT?

`pinMode(13, OUTPUT);`

Zanim będziesz mógł użyć jednego z pinów cyfrowych, musisz poinformować płytke, czy jest to **WEJŚCIE**, czy **WYJŚCIE**. Używamy wbudowanej „funkcji” o nazwie **pinMode()**, aby pin 13 stał się wyjściem cyfrowym. Więcej o wejściach cyfrowych dowiesz się w Projekcie 2.

WAŻNE!

WYJŚCIE CYFROWE

digitalWrite(D13, HIGH);

Kiedy używasz pinu jako WYJŚCIA, możesz na nim ustawić stan WYSOKI (5 woltów) lub NISKI (0 woltów).

OPÓŹNIENIE

delay(2000)

Powoduje, że program czeka przez określony pomiędzy nawiasami {} czas, wyrażony w milisekundach (2000 ms = 2s).

Po upływie tego czasu program będzie kontynuowany od następnej linii kodu.

KOMENTARZ

```
// To jest komentarz  
/* To też jest  
komentarz */
```

Komentarze to świetny sposób na pozostawienie notatek wyjaśniających dlaczego napisałeś to w ten sposób. W komentarzach jednoliniowych używane są dwa ukośniki //, podczas gdy komentarze wieloliniowe zaczynają się od /* i kończą znakiem */.

NOWE POMYSŁY

WYZWANIA KODOWANIA: W sekcji WYZWANIA KODOWANIA znajdziesz sugestie dotyczące zmian w obwodzie lub kodzie, które sprawią, że obwód będzie trudniejszy. Jeśli czujesz się znudzony zadaniami w każdym obwodzie, odwiedź sekcję WYZWANIA KODOWANIA, aby wznieść się na wyższy poziom.

WYZWANIA KODOWANIA

WIDOCZNE MIGOTANIE:

Ekrany komputerów, filmy i światła w Twoim domu migoczą tak szybko, że wydaje się, że są włączone przez cały czas, ale w rzeczywistości migają szybciej, niż ludzkie oko jest w stanie to wykryć.
Zobacz, o ile możesz zmniejszyć czas opóźnienia w programie, zanim dioda będzie wyglądała jakby świeciła cały czas, ale w rzeczywistości nadal będzie migać.

KOD MORSE'A:

Spróbuj dodać i zmienić wartości **delay()** oraz dodać więcej polecen **digitalWrite()**, aby Twój program wyświetlał komunikat w alfabetie Morse'a.

ROZWIAZYWANIE PROBLEMÓW

Wyświetla się błąd podczas wgrywania programu

Najbardziej prawdopodobną przyczyną jest to, że w Arduino IDE wybrano niewłaściwą płytę. Upewnij się, że wybrałeś **TOOLS > BOARD > Arduino/Genuino Uno**.

ROZWIĄZYWANIE PROBLEMÓW

Podczas przesyłania kodu nadal pojawia się błąd

Jeśli jesteś pewien, że wybrałeś właściwą kartę, ale nadal nie możesz przesyłać plików, sprawdź, czy wybrałeś właściwy port szeregowy. Możesz to zmienić w Tools > Serial Port > your_serial_port.

Który port szeregowy jest właściwy?

W zależności od liczby urządzeń podłączonych do komputera możesz mieć kilka aktywnych portów szeregowych. Upewnij się, że wybierasz właściwy. Prostym sposobem ustalenia tego jest sprawdzenie listy portów szeregowych. Odłącz RedBoard od komputera. Spójrz jeszcze raz na listę. Niezależnie od tego, który port szeregowy zniknął z listy, jest to ten, który chcesz wybrać po ponownym podłączeniu płyty do komputera.

Kod został przesłany, ale dioda LED nie włącza się

Diody LED będą działać tylko w jednym kierunku. Spróbuj wyjąć go z deski do krojenia chleba, obrócić o 180 stopni i włożyć ponownie.

Wciąż nie działa?

Przewody połączeniowe niestety mogą się „zepsuć” w wyniku nadmiernego zgięcia. Miedziany drut wewnętrzny może pęknąć, pozostawiając otwarte połączenie w obwodzie. Jeśli masz pewność, że obwód jest prawidłowo podłączony, a kod jest wolny od błędów i został przesłany, ale nadal występują problemy, spróbuj wymienić jeden lub więcej przewodów połączeniowych komponentu, który nie działa.

Koniec obwodu 1A!

Przejdź do obwodu 1B, aby dowiedzieć się o potencjometrach i sygnałach analogowych.



Obwód 1B: Potencjometr

Potencjometry (zwane również „trimpotami” lub „pokrętlami”) to jedne z podstawowych elementów elektronicznych.

Sledząc położenie pokrętła za pomocą RedBoard, możesz regulować głośność, prędkość i mnóstwo innych przydatnych danych wejściowych dla swoich projektów. W tym obwodzie użyjesz potencjometru jako urządzenia wejściowego do kontrolowania prędkości migania diody LED.

CO

POTRZEBĄ



NOWE ELEMENTY

POTENCJOMETR: Potencjometr to 3-pinowy rezystor zmienny. Przy zasilaniu napięciem 5 V środkowy pin generuje napięcie od 0 V do 5 V, w zależności od położenia pokrętła na potencjometrze.



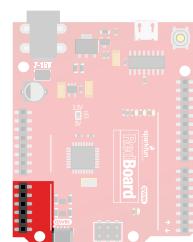
Wewnętrz potencjometru znajduje się pojedynczy rezystor i wycieraczka, która przecina rezystor na dwie części i przesuwa się, aby dostosować stosunek obu połówek.

NOWE POJĘCIA

ANALOGOWY VS. CYFROWY: Żyjemy w świecie analogowym. Istnieje nieskończona liczba kolorów do pomalowania przedmiotu, nieskończona liczba dźwięków, które możemy usłyszeć i nieskończona liczba zapachów, które możemy wyczuć. Wspólnym tematem tych sygnałów analogowych są ich nieskończone możliwości. Sygnały cyfrowe działają w sferze dyskretnej lub skończonej, co oznacza, że istnieje ograniczony zestaw wartości, jakie mogą przyjmować. Dioda LED z poprzedniego obwodu miała tylko dwa stany, w jakich mogła istnieć: WŁĄCZONA lub WYŁĄCZONA, po podłączeniu do wyjścia cyfrowego.

WEJŚCIA ANALOGOWE: Do tej pory zajmowaliśmy się tylko wyjściami. RedBoard posiada również wejścia. Zarówno wejścia, jak i wyjścia mogą być analogowe lub cyfrowe. W oparciu o naszą poprzednią definicję analogu i cyfry oznacza to, że wejście analogowe może wykrywać szeroki zakres wartości w porównaniu z wejściem cyfrowym, które może wykrywać tylko dwie wartości lub stany.

Być może zauważysz, że niektóre piny są oznaczone jako **Digital**, a inne jako **Analog** na Twojej płytce. Jest tylko sześć pinów, które pełnią funkcję wejść analogowych; są one oznaczone jako A0 – A5.

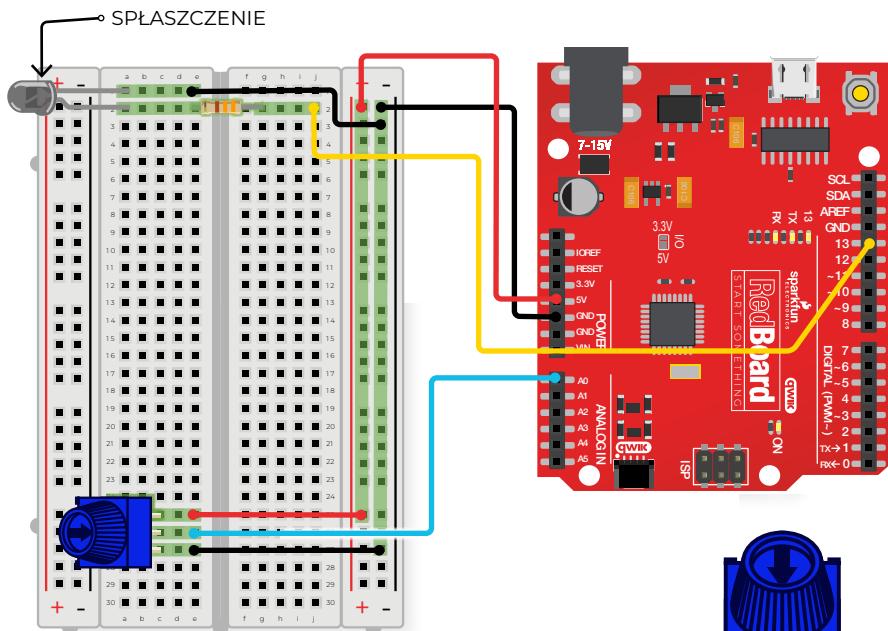


DZIELNIK NAPIĘCIA

ROZDZIELNIKI NAPIĘCIA to proste obwody, które zamieniają część napięcia na mniejsze napięcie za pomocą dwóch rezystorów. Potencjometr to rezystor zmienny, który można wykorzystać do utworzenia regulowanego dzielnika napięcia. Pokrętło w położeniu środkowym oznacza, że napięcie wyjściowe będzie stanowić połowę napięcia wejściowego. Dzielniki napięcia zostaną omówione bardziej szczegółowo przy następnym obwodzie.

JAK PODŁĄCZAĆ?

GOTOWY, ABY ZACZĄĆ WSZYSTKO PODŁĄCZAĆ? Sprawdź schemat obwodu i tabelę połączeń poniżej, aby zobaczyć, jak wszystko jest podłączone.



NOWE POJĘCIE

POTENCJOMETRY nie są spolaryzowane i można je instalować w dowolnym kierunku. Należy pamiętać, że zamiana pinów 5 V i GND odwróci swoje zachowanie.

POŁĄCZENIA:

◆ PŁYTKA REDBOARD

■ PŁYTKA BREADBOARD

◆ 5V to □ 5V

34

◆ GND to □ GND (-)

GRB (1)

◆ A0 to F2

1

100

■ A1(-) to ■ A2(+)

**REZYSTR 330Ω
(POMARAŃCZOWY,
POMARAŃCZOWY
BRAZOWY)**

E2 to G2

POTENCIOMETR

■ C25 + ■ C26 + ■ C27

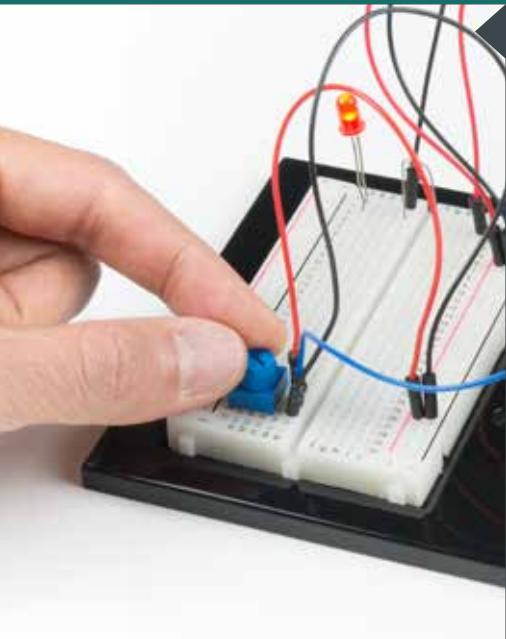
Otwórz Arduino IDE

Podłącz płytke RedBoard do portu USB Twojego komputera.

1 Otwórz szkic:

File > Examples >SIK-Guide-Code-master > **CIRCUIT_1B-POTENTIOMETER**

2 Kliknij **UPLOAD** aby załadowac program na płytke



Co powinieneś zobaczyć

Powinieneś zobaczyć, że dioda LED migła szybciej lub wolniej, zgodnie z ustawieniem potencjometra. Opóźnienie pomiędzy kolejnymi błyskami będzie się zmieniać w zależności od położenia pokrętła. Jeśli to nie działa, upewnij się, że poprawnie zmontowałeś obwód, zweryfikowałeś i przesłałeś kod na swoją płytke. Jeśli to nie zadziała, zobacz sekcję Rozwiązywanie problemów.

CO ROBI PROGRAM?

- 1 Odczytaj pozycje potencjometru w zakresie 0 do 1023 i zapisz ją w zmiennej **potPosition**
- 2 Zaświeć LED
- 3 Poczekaj od 0 do 1023 milisekund - w zależności od zawartości zmiennej **potPosition**
- 4 Zgaś LED
- 5 Poczekaj od 0 do 1023 milisekund - w zależności od zawartości zmiennej **potPosition**
- 6 Powtórz.

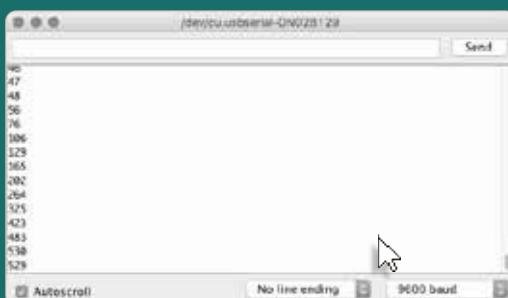
ARDUINO WSKAZOWKI

MONITOR SZEREGOWY ARDUINO: Monitor szeregowy to jedna z wielu wspaniałych funkcji dostępnych w Arduino IDE. Podczas pracy z systemami osadzonymi pomaga zobaczyć i zrozumieć wartości, z którymi program próbuje pracować i może być potężnym narzędziem do debugowania, gdy natkiesz się na problemy, w których kod nie zachowuje się tak, jak tego oczekiwaneś.

Ten obwód stanowi wprowadzenie do monitora szeregowego, pokazując, jak wyświetlić na nim wartości z potencjometru. Aby zobaczyć te wartości, kliknij przycisk Serial Monitor, znajdujący się w prawym górnym rogu IDE w najnowszych wersjach. Możesz także wybrać z menu **Tools > Serial Monitor**.



*Serial Monitor button
in the upper-right of
the Arduino IDE.*



*Serial Monitor printout
and baud-rate menu.*

Powinieneś zobaczyć wydrukowane wartości numeryczne na monitorze. Obracanie potencjometru zmienia wartość oraz opóźnienie pomiędzy kolejnymi wydrukami. Jeśli masz problemy z odczytaniem wartości, upewnij się, że wybrałeś 9600 bodów i sprawdź automatyczne przewijanie.

WAŻNE

ZMIENNE TYPU CAŁKOWITEGO:

```
int potPosition;
```

Zmienna jest symbolem zastępczym wartości, które mogą ulec zmianie w kodzie. Musisz wprowadzić lub „zadeklarować” zmienne przed ich użyciem. Tutaj deklarujemy zmienną zwaną **potPosition**; typu **int** (integer). Więcej typów zmiennych omówimy w późniejszych obwodach. Nie zapominaj, że w nazwach zmiennych rozróżniana jest wielkość liter!

KOD

INICJOWANIE PORTU SZEREGOWEGO:
Serial.begin(9600);

Poleceń dotyczących portu szeregowego można używać do wysyłania i odbierania danych z komputera. Ta linia kodu mówi płytce, że chcemy „rozpocząć” komunikację z komputerem w ten sam sposób, w jaki powiedzielibyśmy „Cześć”, aby rozpocząć rozmowę. Zauważ, że szybkość transmisji 9600 jest taka sama, jak ta, którą wybraliśmy na monitorze. Jest to prędkość, z jaką komunikują się oba urządzenia i musi ona być zgodna po obu stronach.

WEJŚCIE ANALOGOWE:
potPosition =
analogRead(A0);

Do odczytania wartości na pinie analogowym używamy funkcji **analogRead()**. Funkcja **analogRead()** pobiera jeden parametr, pin analogowy, którego chcesz użyć, w tym przypadku **A0**, i zwraca liczbę z zakresu od 0 (0 woltów) do 1023 (5 woltów), która jest przypisana do zmiennej **potPosition**.

PISANIE NA PORT SZEREGOWY:
Serial.
println(potPosition);

Jest to linia, która wypisuje na monitorze szeregowym wartość potencjometru. Pobiera zmienną **potPosition** i wypisuje w pętli **loop()** wartość, jaką w danym momencie osiąga. **In** na końcu **println()** informuje monitor, aby przeszedł do nowej linii na końcu każdej wartości, w przeciwnym razie wszystkie wartości byłyby wyświetlane razem w jednym wierszu. Spróbuj usunąć **In**, aby zobaczyć, co się stanie.

WYZWANIA KODOWANIA

ZMIANA ZAKRESU: Spróbuj pomnożyć, podzielić lub dodać jakąś wartość do odczytu z czujnika, aby móc zmienić zakres opóźnienia w kodzie. Na przykład, czy można pomnożyć odczyt czujnika, aby opóźnienie wynosiło od 0–2046 zamiast 0–1023?

DODAJ WIĘCEJ DIOD LED: Dodaj więcej diod LED do swojego obwodu. Nie zapomnij o rezystorach ograniczających prąd. Będziesz musiał zadeklarować nowe piny w swoim kodzie i ustawić je wszystkie na **OUTPUT**.

Spróbuj sprawić, aby poszczególne diody LED migły z różną szybkością, zmieniając opóźnienia każdej z nich za pomocą mnożenia lub dzielenia.

ROZWIĄZYWANIE PROBLEMÓW

Odczyt z potencjometru zawsze wynosi 0 lub 1023	Upewnij się, że piny 5 V, A0 i GND są prawidłowe podłączony do trzech pinów potencjometru. To jest łatwo pomylić przewód z rzeczywistym pinem potencjometru.
Brak wartości lub losowe znaki w monitorze portu szeregowego	Upewnij się, że wybrałeś prawidłową prędkość transmisji, 9600 . Upewnij się także, że korzystasz z prawidłowego portu szeregowego. Ten sam port szeregowy, którego używasz podczas przesyłania kodu na płytke, jest tym samym portem szeregowym, którego używasz do wypisywania wartości na monitorze szeregowym.

Ukończyłeś
Obwód 1B!

Przejdź do obwodu 1C, aby poznać fotorezystory i konwersję sygnału analogowego na cyfrowy.



Obwód 1C: Fotorezystor

W obwodzie 1B należy użyć potencjometru, który zmienia rezystancję w oparciu o obrót pokrętła.

W tym obwodzie będziesz używać fotorezystora, który zmienia rezystancję w zależności od ilości światła padającego na niego. Za pomocą tego czujnika możesz stworzyć prostą lampkę nocną, która włącza się, gdy w pomieszczeniu robi się ciemno, i wyłącza, gdy jest jasno.

POTRZEBĄ:



NOWE ELEMENTY

FOTOREZYSTORY są światłoczułymi rezystorami zmiennymi. W miarę padania większej ilości światła na główce czujnika, rezystancja pomiędzy jego dwoma zaciskami maleje. Stanowią łatwy w użyciu komponent w projektach wymagających wykrywania światła otoczenia.

NOWE POJĘCIA

KONWERSJA SYGNAŁU ANALOGOWEGO NA CYFROWY:

Aby RedBoard mógł odczytywać sygnały analogowe, musimy przekazać je przez przetwornik analogowo-cyfrowy (ADC). Wszystkie sześć wejść analogowych (A0-A5) w ostatnim obwodzie wykorzystuje przetwornik ADC. Piny te próbują sygnały analogowy i tworzą sygnał cyfrowy łatwy do interpretacji przez mikrokontroler.

Rozdziercość tego sygnału opiera się na rozdziercości przetwornika ADC. W przypadku RedBoard rozdziercość ta wynosi 10 bitów. Przy 10-bitowym przetworniku ADC otrzymujemy $2^{10} = 1024$ możliwych wartości, dlatego sygnał analogowy może zmieniać się w zakresie od 0 do 1023.

DZIELNIKI NAPIĘCIA CIĄG DALSZY:

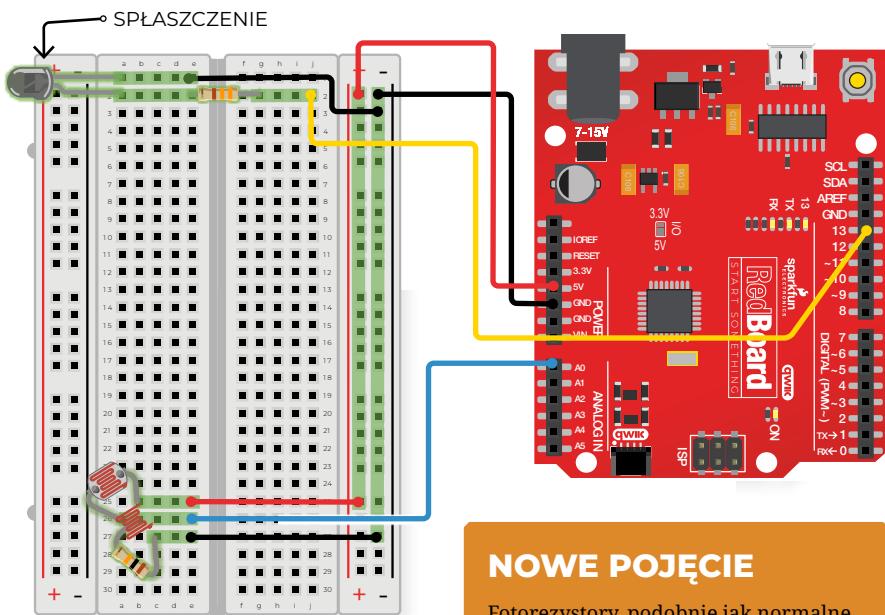
Ponieważ RedBoard nie może bezpośrednio interpretować rezystancji (raczej odczytuje napięcie), musimy użyć dzielnika napięcia, aby wykorzystać nasz fotorezystor, część, która nie wytwarza napięcia wyjściowego. Opór fotorezystora zmienia się, gdy robi się ciemniejszy lub jaśniejszy. To zmienia lub „dzieli” napięcie przechodzące przez obwód dzielnika. To podzielone napięcie jest następnie odczytywane na przetworniku analogowo-cyfrowym wejścia analogowego. Równanie dzielnika napięcia:

$$V_{out} = V_{in} \cdot \frac{R_2}{R_1 + R_2}$$

zakłada, że znasz trzy wartości powyższego obwodu: napięcie wejściowe (V_{in}) i wartości obu rezystorów (R_1 i R_2). Jeśli R_1 ma wartość stałą (rezystor), a R_2 się zmienia (fotorezystor), wielkość napięcia mierzona na pinie V_{out} również będzie się zmieniać.

JAK PODŁĄCZAĆ?

GOTOWY, ABY ZACZĄĆ WSZYSTKO PODŁĄCZAĆ? Sprawdź schemat obwodu i tabelę połączeń poniżej, aby zobaczyć, jak wszystko jest podłączone.



NOWE POJĘCIE

Fotorezystory, podobnie jak normalne rezystory nie są spolaryzowane i mogą być instalowane w dowolnym kierunku.

POŁĄCZENIA:

◆ PŁYTKA REDBOARD

■ PŁYTKA BREADBOARD

- | | | |
|------------------|--------------------|------------------|
| ◆ 5V to ■ 5V(+) | ◆ GND to ■ GND (-) | |
| ◆ D13 to ■ J2 | ◆ A0 to ■ E26 | ■ E1 to ■ GND(-) |
| ■ E25 to ■ 5V(+) | ■ E27 to ■ GND(-) | |

LED

■ A1(-) to ■ A2(+)

REZYSTOR 330Ω
(PMARAŃCZOWY, POMARAŃCZOWY,
BRĄZOWY)

■ E2 to ■ G2

REZYSTOR 10kΩ
(BRĄZOWY, CZARNY,
POMARAŃCZOWY)

■ B26 to ■ C27

FOTOREZYSTOR

■ A26 to ■ B25

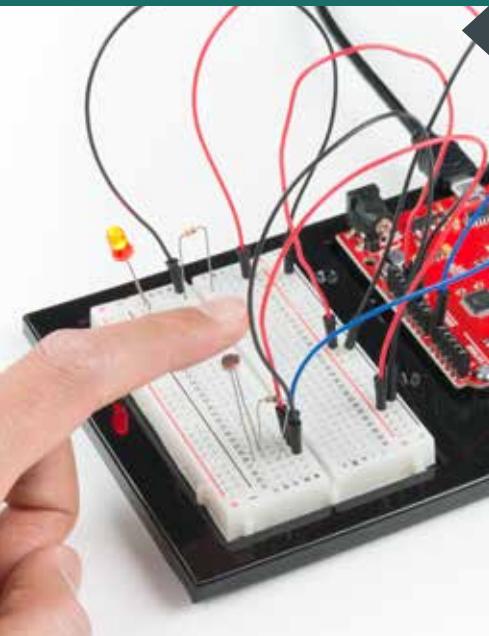
Otwórz Arduino IDE

Podłącz płytę RedBoard do portu USB Twojego komputera

1 Otwórz szkic:

File > Examples > SIK-Guide-Code-master > **CIRCUIT_1C-PHOTORESISTOR**

2 Wybierz **UPLOAD** aby skompilować i wgrać szkic na płytę



CO POWINIENEŚ ZOBACZYĆ?

Program przechowuje odczytany poziom światła w zmiennej. Za pomocą instrukcji if/else wartość zmiennej jest porównywana z progiem. Jeśli zmienna jest powyżej progu (jest jasno), wyłącza diodę LED. Jeśli zmienna jest poniżej progu (jest ciemno), włącza diodę LED.

Otwórz monitor szeregowy w Arduino. Wartość napięcia z fotorezystora powinna być drukowana co jakiś czas. Gdy wartość fotorezystora spadnie poniżej progu, dioda LED powinna się zaświecić (w celu sprawdzenia fotorezystora można zasłonić palcem).

NOWE POJĘCIE

POZIOMY OSŁWIETLENIA: Jeśli pomieszczenie, w którym się znajdujesz, jest bardzo jasne lub ciemne, być może będziesz musiał zmienić wartość zmiennej **threshold** w kodzie, aby włączać i wyłączać lampkę nocną. Instrukcje znajdziesz w sekcji Rozwiązywanie problemów.

CO ROBI PROGRAM?

1 Zapisuje poziom oświetlenia w zmiennej **photoresistor**

2 Jeśli wartość **photoresistor** jest powyżej **threshold** (jest jasno), wyłącza LED

3 Jeśli zaś wartość **photoresistor** jest poniżej **threshold** (jest ciemno), włącza LED

KOD

WARUNKI IF/ELSE

```
if(logic statement){  
//run if true  
}  
  
else{  
//run if false  
}
```

OPERATORY LOGICZNE

```
(photoResistor <  
threshold)
```

Instrukcja **if else** pozwala reagować na świat, uruchamiając jeden zestaw kodu, gdy instrukcja logiczna w nawiasach okrągłych jest prawdziwa, a drugi zestaw kodu, gdy instrukcja logiczna jest fałszywa. Na przykład w tym szkicu zastosowano instrukcję **if**, aby włączyć diodę LED, gdy jest ciemno, oraz instrukcję **else**, aby wyłączyć diodę LED, gdy jest jasno.

Programiści używają instrukcji logicznych do przekładania na kod zdarzeń zachodzących w prawdziwym świecie. Instrukcje logiczne używają operatorów logicznych, takich jak „równy” `==`, „większy niż” `>` i „mniejszy niż” `<`, do dokonywania porównań. Jeśli porównanie jest prawdziwe (np. `4 < 5`), wówczas stwierdzenie logiczne jest prawdziwe. Jeśli porównanie jest fałszywe (np. `5 < 4`), wówczas stwierdzenie logiczne jest fałszywe. W tym przykładzie zadano pytanie, czy wartość zmiennej **photoResistor** jest mniejsza niż zmiennej **threshold**.

WYZWANIA KODOWANIA

MRUGANIE NA MACHANIE: W tej chwili instrukcja `if` włącza diodę LED, gdy robi się ciemno, ale możesz także używać czujnika światła jak przycisku bezdotykowego.

Spróbuj użyć funkcji `digitalWrite()` i `delay()`, aby dioda LED migała według wzoru, gdy poziom światła spada, a następnie skalibruj zmienną progową w kodzie tak, aby mruganie uruchamiało się po machnięciu ręką nad czujnikiem.

ZAMIEŃ REZYSTOR 10 kΩ NA DIODĘ LED: Zmień obwód, zastępując rezystor 10 kΩ diodą LED (noga ujemna powinna być podłączona do GND). Co się stanie, gdy położysz palec na fotorezystorze? To świetny sposób, aby zobaczyć, jak prawo Ohma działa, poprzez wizualizację wpływu zmiany rezystancji na prąd przepływający przez diodę LED.

ROZWIĄZYWANIE PROBLEMÓW

Na monitorze szeregowym nic się nie drukuje

Spróbuj odłączyć kabel USB i podłączyć go ponownie. W Arduino IDE przejdź do **Tools > Port** i upewnij się, że wybrałeś właściwy port.

Światło nigdy się nie włącza lub zawsze pozostaje włączone

Uruchom monitor szeregowy w Arduino. Spójrz na wartość odczytaną przez fotorezystor w jasnym pomieszczeniu (np. 915). Zakryj fotorezystor lub wyłącz światło. Następnie spójrz na nową wartość odczytywaną przez fotorezystor (np. 550). Ustaw próg pomiędzy tymi dwiema liczbami (np. 700), tak aby odczyt znajdował się powyżej progu, gdy światła są włączone i poniżej progu, gdy światła są wyłączone.

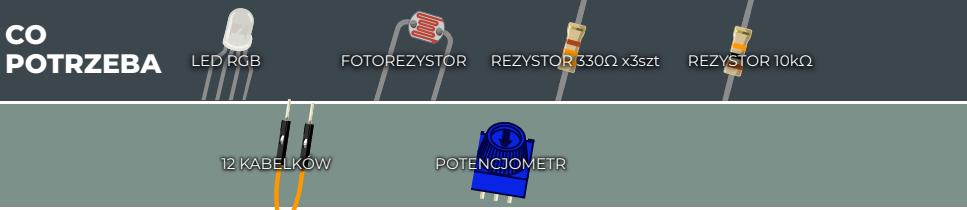
Ukończyłeś obwód 1C!

Przejdź do obwodu 1D aby nauczyć się o diodach RGB funkcjach oraz modulacji szerokości impulsu.



Obwód 1: Nocna Lampka RGB

W tym obwodzie przeniesiesz koncepcję oświetlenia nocnego na wyższy poziom dodając diodę LED RGB, czyli trzy różnokolorowe diody elektroluminescencyjne (LED) wbudowane w jedną obudowę. RGB oznacza czerwony, zielony i niebieski, a te trzy kolory można łączyć, aby stworzyć dowolny kolor tęczy!



NOWE ELEMENTY

Dioda LED RGB: Dioda LED RGB to w rzeczywistości trzy małe diody LED — jedna czerwona, jedna zielona i jedna niebieska — umieszczone w normalnej obudowie diody LED. W przypadku tej diody wszystkie wewnętrzne diody LED mają jedną wspólną elektrodę, więc w sumie są cztery nogi. Aby włączyć jeden kolor, upewnij się, że masa jest podłączona, a następnie zasil jedną z nóg tak samo, jak zwykłą diodę LED. Nie zapomnij o rezystorach ograniczających prąd. Jeśli włączysz więcej niż jeden kolor na raz, kolory zaczyną się ze sobą mieszać, tworząc nowy kolor.



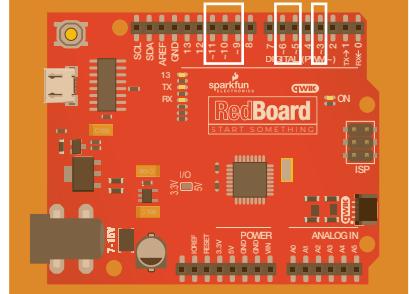
NOWE POJĘCIA

WYJŚCIE ANALOGOWE (MODULACJA

SZEROKOŚCI IMPULSU: Polecenie **digitalWrite()** może włączyć (5 V) lub wyłączyć piny (0 V), ale co, jeśli chcesz wyprowadzić napięcie 2,5 V? Polecenie **analogWrite()** może wyprowadzić napięcie 2,5 V poprzez szybkie włączanie i wyłączanie pinów,

NOWE POJĘCIA

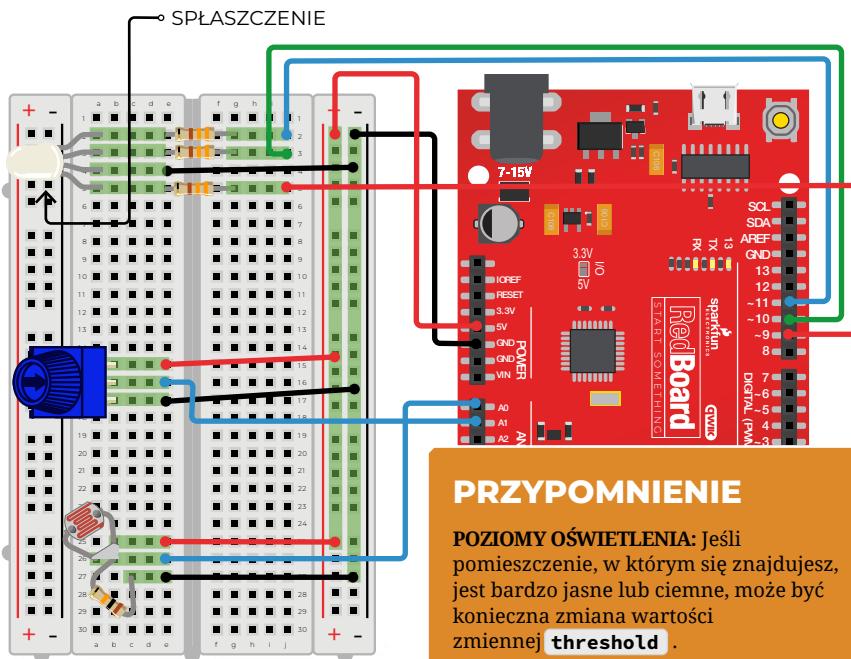
WYJŚCIA PWM: Tylko kilka pinów na płycie RedBoard ma obwody potrzebne do szybkiego włączania i wyłączania dla PWM. Są to piny 3, 5, 6, 9, 10 i 11. Każdy pin PWM jest oznaczony na płytce ~.



tak że jest on włączony tylko przez 50 procent czasu (50% z 5 V to 2,5 V). W ten sposób można wytworzyć dowolne napięcie w zakresie od 0 do 5 V. Jest to tak zwana modulacja szerokości impulsu (PWM). Dzięki temu ożemy tworzyć wiele różnych kolorów na diodzie LED RGB.

JAK PODŁĄCZYĆ

GOTOWY, ABY ZACZĄĆ WSZYSTKO PODŁĄCZYĆ? Sprawdź schemat obwodu i tabelę połączeń poniżej, aby zobaczyć, jak wszystko jest podłączone.



PRZYPOMNIENIE

POZIOMY OŚWIETLENIA: Jeśli pomieszczenie, w którym się znajdujesz, jest bardzo jasne lub ciemne, może być konieczna zmiana wartości zmiennej `threshold`.

Połączenia:

◆ PŁYTKA REDBOARD

■ PŁYTKA BREADBOARD

KARELKU

- ◆ 5V to ■ 5V(+)
 - ◆ GND to ■ GND (-)
 - ◆ D9 to ■ J5
 - ◆ D10 to ■ J3
 - ◆ D11 to ■ J2
 - ◆ A0 to ■ E26
 - ◆ A1 to ■ E16
 - E15 to ■ 5V(+)
 - E17 to GND(-)
 - E4 to ■ GND(-)
 - E25 to ■ 5V(+)
 - E27 to ■ GND (-)

LED RGB

- A5(RED) + ■ A4(GND) + ■ A3(GREEN) + ■ A2(BLUE)

**REZYSTOR 330Ω
OMARAŃCZOWY,
BRAZOWY)**

- E2 to ■ G2 ■ E3 to ■ G3 ■ E5 to ■ G5

**REZYSTOR 10kΩ
(BRĄZOWY, CZARNY,
POMARAŃCZOWY)**

- B26 to C27

FOTOREZYSTOR

- A26 to ■ B25

POTENCIOMETR

- B15 + ■ B16 + ■ B17

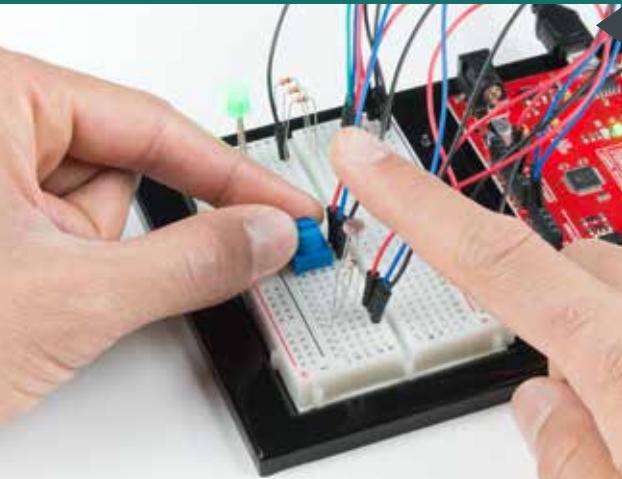
Otwórz Arduino IDE

Podłącz płytke RedBoard do portu USB Twojego komputera.

1 Otwórz szkic:

File > Examples > SIK-Guide-Code-master > **SIK_CIRCUIT_1D-RGB NIGHT LIGHT**

2 Kliknij **UPLOAD** aby skompilować i wgrać program na płytę



CO ZOBACZE?

Ten szkic nie różni się od poprzedniego. Odczytuje wartość z fotorezystora, porównuje ją z wartością progową i odpowiednio włącza lub wyłącza diodę LED RGB. Tym razem jednak dodaliśmy z powrotem potencjometr do obwodu. Po przekręceniu potencjometru powinieneś zobaczyć zmianę koloru diody LED RGB w zależności od wartości potencjometru.

CO ROBI PROGRAM

- 1 Zapisz wartość poziomu światła (A0) w zmiennej **photoresistor**
- 2 Zapisz wartość z potencjometru (A1) w zmiennej **potentiometer**
- 3 Jeśli poziom światła jest większy niż **threshold** wywołaj funkcję włączającą diode RGB
- 4 Jeśli poziom światła jest mniejszy niż **threshold** wywołaj jedną z funkcji zaświecającą diodę RGB
- 5 Jeśli wartość **potentiometer** wynosi pomiędzy 0 i 150, włącz kolor czerwony.
- 6 Jeśli wartość **potentiometer** wynosi pomiędzy 151 i 300, włącz kolor pomarańczowy.
- 7 Jeśli wartość **potentiometer** wynosi pomiędzy 301 i 450, włącz kolor żółty.
- 8 Jeśli wartość **potentiometer** wynosi pomiędzy 451 i 600, włącz kolor zielony.
- 9 Jeśli wartość **potentiometer** wynosi pomiędzy 601 i 750, włącz kolor cyjan.
- 10 Jeśli wartość **potentiometer** wynosi pomiędzy 751 i 900, włącz kolor niebieski.
- 11 Jeśli wartość **potentiometer** jest większa niż 900, włącz kolor magenta.

KOD

WYJŚCIE ANALOGOWE (PWM)

```
analogWrite(RedPin, 100);
```

Funkcja **analogWrite()** wyprowadza na pin napięcie od 0 do 5 V. Funkcja dzieli zakres od 0 do 5 V na 255 małych kroków. Należy pamiętać, że w tym kodzie nie włączamy diody LED do pełnej jasności (255), aby światło nocne nie było zbyt jasne. Zmień te wartości i zobacz, co się stanie.

ZAGNIEŻDŻONE WARUNKI

```
if(logic statement){  
    if(logic statement){  
        } }
```

Zagnieżdżona instrukcja **if** to jedna lub więcej instrukcji **if** „zagnieżdżonych” w innej instrukcji **if**. Jeśli nadrędzona instrukcja **if** jest prawdziwa, kod sprawdza każdą zagnieżdżoną instrukcję **if** i wykonuje każdą z nich, która jest prawdziwa. Jeśli nadrędzona instrukcja **if** jest fałszywa, żadna z zagnieżdżonych instrukcji nie zostanie wykonana.

WIĘCEJ OPERATORÓW

LOGICZNYCH

```
(potentiometer > 0 &&  
potentiometer <= 150)
```

Te instrukcje **if** sprawdzają dwa warunki za pomocą operatora AND **&&**. W tej linii instrukcja **if** będzie prawdziwa tylko wtedy, gdy wartość zmiennej **potentiometer** jest większa niż 0 **ORAZ** jeśli wartość jest mniejsza lub równa 150. Używając **&&**, program pozwala diodzie LED mieć wiele stanów kolorów.

DEFINIACJA FUNKCJI

```
void function_name(){ }
```

To jest definicja prostej funkcji. Kiedy programiści chcą wielokrotnie używać wielu linii kodu, piszą funkcję. Kod w nawiasach klamrowych „wykonuje się” za każdym razem, gdy funkcja jest „wywoływana” w programie głównym. Każdy z kolorów diody LED RGB jest zdefiniowany w funkcji.

WYWОŁANIE FUNKCJI

```
function_name();
```

Ta linia „wywołuje” utworzoną funkcję. W dalszej części dowiesz się, jak tworzyć bardziej skomplikowane funkcje, które pobierają dane z programu głównego (te fragmenty danych nazywane są parametrami).

WYZWANIA KODOWANIA

DODAJ WIĘCEJ KOLORÓW: Dzięki diodzie LED RGB możesz stworzyć o wiele więcej kolorów. Użyj funkcji `analogWrite()`, aby połączyć różne wartości koloru czerwonego, zielonego i niebieskiego, aby uzyskać jeszcze więcej kolorów. Możesz podzielić wartość potencjometru i utworzyć więcej zagnieźdzonych instrukcji if, aby uzyskać więcej kolorów podczas kręcenia pokrętlem.

MIGANIE WIELOKOLOROWE: Spróbuj użyć opóźnień i funkcji wielu kolorów, aby dioda LED RGB przełączała się między wieloma kolorami, gdy jest ciemno.

ZMIENЬ PRÓG: Spróbuj ustawić zmienną progową, odczytując wartość potencjometru. Obracając potencjometr, możesz następnie zmienić poziom progu i dostosować oświetlenie nocne do różnych pomieszczeń.

WYGASZANIE DIODY LED: Użyj funkcji `analogWrite()`, aby dioda LED pulsowała delikatnie lub płynnie przechodziła między kolorami.

ROZWIĄZYwanie problemów

Dioda nigdy się nie zaświeca lub nie gaśnie

Otwórz monitor szeregowy i upewnij się, że fotorezystor zwraca wartości z zakresu od 0 do 1023. Zakryj fotorezystor; wartości powinny się zmienić. Jeśli się nie zmieniają, sprawdź obwód. Upewnij się, że zmienna progowa mieści się pomiędzy wartością odczytywaną przez fotorezystor, gdy jest jasny, a wartością, gdy jest ciemno (np. jasny = 850, ciemny = 600, próg = 700).

Dioda nie świeci kolorami jakich oczekuję

Upewnij się, że wszystkie trzy piny sterujące diodą LED RGB są ustawione na OUTPUT, używając polecenia `pinMode()` w sekcji ustawień kodu. Następnie upewnij się, że każda nóżka diody LED jest prawidłowo podłączona.

Nic nie jest wyświetlane na Serial Monitorze

Spróbuj odłączyć kabel USB i podłączyć go ponownie. W Arduino IDE przejdź do Narzędzia > Port i wybierz odpowiedni port.

Ukończyłeś Obwód 1D!

Przejdź do projektu 2 aby przy pomocy buzzera zrobić trochę hałasu!

MIGANIE LED

ODCZYT Z POTENCJOMETRU

ODCZYT Z FOTOREZYSTORA

LAMPKA NOCNA RGB

A

B

C

D