

Wprowadzenie do obliczeń fizycznych

Rozmawialiśmy o tym, jak programować w Pythonie. Nadszedł czas, aby wykorzystać nasze nowo nabyte umiejętności Pythona, aby zacząć robić rzeczy w prawdziwym świecie. Nazywamy to komputerem fizycznym - zmuszaniem komputera do interakcji z otaczającym cię światem! Naszym zdaniem trudniej jest poznać oprogramowanie (Python) niż sprzęt. Dlatego ta książka skupia się głównie na nauce programowania komputerów w Pythonie. Ale teraz nadszedł czas, aby nauczyć robić komputery coś z Pythonem. W tej części podłączamy różne czujniki i silniki do komputera Raspberry Pi. Chociaż napięcia (3,3 V i 5 V) stosowane w tych komputerach nie są niebezpieczne dla ludzi, nieprawidłowe podłączenie może spowodować spalenie komputera lub czujników. Z tego powodu przestrzegaj wytrwale tych dwóch zasad:

Zasada 1: Wyłącz zasilanie przed podłączeniem lub zmianą jakichkolwiek przewodów.

Zasada 2: Dokładnie sprawdź połączenia, zwłaszcza połączenia zasilania, zasilania i uziemienia. Są to najważniejsze przewody do sprawdzenia.

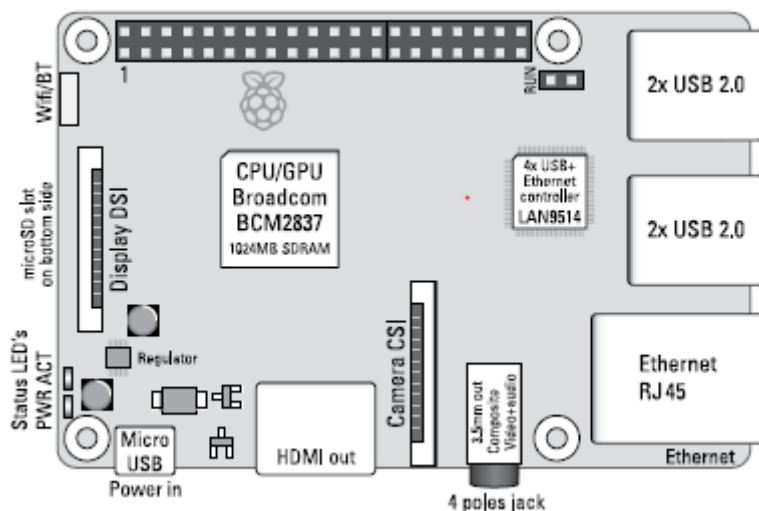
Obliczenia fizyczne są zabawne

Jednym z powodów, dla których chcemy, abyś dowiedział się o komputerach fizycznych, jest to, że małe komputery wykonujące fizyczne czynności (zwykle nazywane systemami wbudowanymi) są wszędzie wokół ciebie. I mamy na myśli wszędzie. Idź do swojej kuchni. Rozejrzyj się. Twoja lodówka ma komputer, może dwa lub trzy, jeśli ma wyświetlacz. Twój blender ma komputer. Twój piekarnik jest wyposażony w komputer. Twoja kuchenka mikrofalowa jest wyposażona w komputer. Jeśli używasz w domu światła Phillips Hue, Twoje żarówki mają komputer. Twój samochód będzie miał w pojeździe ponad 20 komputerów. Jeszcze jeden przykład. Co powiesz na skromny toster? Jeśli masz przycisk „Bajgiel” lub wyświetlacz na swoim tosterze, masz tam komputer. Dlaczego? Dlaczego w twoim domu jest tyle komputerów? Ponieważ zbudowanie wszystkich gadżetów przy użyciu komputera jest znacznie tańsze niż zaprojektowanie specjalnego sprzętu. Czy wiesz, że możesz kupić komputery (luzem) za około 0,15 USD? Komputery są wszędzie. Większość z tych komputerów jest znacznie prostsza, wolniejsza i zawiera znacznie mniej pamięci RAM (rodzaj pamięci) niż przeciętny komputer. Komputer PC może mieć około 4–16 lub więcej GB (to jest gigabajty, a 1 GB to około 1 miliard bajtów), ale komputer, na którym działa toster, ma prawdopodobnie tylko około 100 bajtów. Jest to różnica ponad 10 000 000 razy większa od ilości pamięci RAM. Nawiasem mówiąc, możesz myśleć, że jeden bajt jest równy jednemu znakowi w języku angielskim. W większości krajów azjatyckich jeden znak to dwa bajty. Komputery są więc wszędzie. Ale interesujące jest to, że wszystkie te małe komputery wykonują obliczenia fizyczne. Wyczuwają i wchodzą w interakcje z otoczeniem. Komputer lodówki sprawdza, czy ma odpowiednią temperaturę, a jeśli nie, włącza maszynę chłodzącą, zwracając uwagę na to, co robi, aby zminimalizować ilość zużywanej energii elektrycznej. Piec aktualizuje wyświetlacz na panelu przednim, monitoruje przyciski i pokrętła oraz kontroluje temperaturę, dzięki czemu masz dobrą lasagne na obiad. Wszystkie te interakcje i kontrolery nazywane są obliczeniami fizycznymi.

Co to jest Raspberry Pi?

Moglibyśmy użyć jednego z tych bardzo małych komputerów, ale jego funkcjonalność jest ograniczona w porównaniu z twoim komputerem PC, więc pójdziemy na kompromis i użyjemy Raspberry Pi, komputera za 35 dolarów, który ma ogromną ilość sprzętu i oprogramowania dostępnego do użytku, zwłaszcza w Pythonie). Jest bardziej złożony niż toster, ale jest dużo prostszy nawet niż komputer używany w telewizorze. Raspberry Pi to popularny SBC (komputer jednopłytkowy), który istnieje od około 2012 roku. Został stworzony przez Fundację Raspberry Pi w celu nauczania podstaw nauk ścisłych i inżynierii w szkołach na całym świecie. Okazał się szalenie popularny i sprzedał ponad 19 milionów komputerów

na całym świecie. Dostępnych jest wiele innych modeli Raspberry Pi (od Raspberry Pi Zero za 5 USD do nowego Raspberry Pi 3B, którego będziemy używać w tej książce). Aby wyjaśnić niektóre technologie, z którymi mamy do czynienia na co dzień, porozmawiajmy o głównych blokach sprzętowych tego komputera. Pamiętaj, że Twój smartfon ma w środku komputery, które pod względem budowy są bardzo podobne do Raspberry Pi. Rysunek przedstawia główne bloki komputera:



Złącze GPIO: Jest to złącze wtykowe wejścia-wyjścia ogólnego przeznaczenia. W pozostałej części tego minibooka będziemy często używać tego złącza.

CPU/GPU: jednostka centralna/procesor graficzny (dla ekranu). Ten blok jest mózgiem sprzętu i mówi wszystkim innym, co robić. Twoje programy w Pythonie będą uruchamiane przez ten blok.

USB: Są to standardowe porty USB (uniwersalna magistrala szeregową), te same interfejsy, które można znaleźć w dużych komputerach. Istnieje wiele urządzeń, które można podłączyć do portu USB, tak jak na komputerze. Do tych portów podłączysz mysz i klawiaturę.

Ethernet: Podobnie jak interfejs Ethernet w komputerze. Łączy się z siecią za pomocą przewodów.

Wi-Fi: Ten blok nie jest pokazany na schemacie, ale jest bardzo przydatny. Dzięki WiFi nie musisz ciągać kabli, aby rozmawiać z Internetem.

Wyjście HDMI: Do tego portu podłączasz monitor lub telewizor.

Gniazdo audio: dźwięk i wideo kompozytowe (stary standard).

Inne porty: Trzy bardziej interesujące porty na płycie to:

- Micro USB: To jest twój zasilacz 5V.
- Camera CSI: Służy do połączenia kablem taśmowym z kamerą Raspberry Pi.
- Display DSI: Służy do szybkich połączeń z różnymi niestandardowymi wyświetlaczami.

Sprawianie, że komputer robi różne rzeczy

Aby nasz komputer mógł robić i wyczuwać rzeczy poza ekranem komputera i klawiaturą, potrzebujemy komputera i jednej z dwóch innych rzeczy - czujnika lub elementu wykonawczego. Czujnik to mały element elektroniczny, który może wykryć coś w środowisku (na przykład temperaturę lub wilgotność), a siłownik to wymyślne słowo określające silnik lub kabel, który robi rzeczy w prawdziwym świecie. W

dalszej części poznamy niezbędne składniki naszego pierwszego fizycznego projektu obliczeniowego, czyli włączanie i wyłączanie diody LED. Jest to fizyczna wersja programu „Hello World”, którą wszyscy wykonujemy, gdy uczymy się oprogramowania. Migająca dioda LED, nadchodzimy! Wyjdź teraz i kup zestaw startowy Raspberry Pi (zawiera zasilacz, system operacyjny i obudowę) i skonfiguruj go przed kontynuowaniem. Zalecamy chwycenie myszy, klawiatury i monitora, aby przeprowadzić konfigurację dla początkujących, ale bardziej zaawansowani użytkownicy mogą chcieć użyć SSH (Secure SHell), aby przeprowadzić konfigurację bez głowy. Ponownie, najlepszym miejscem do rozpoczęcia jest www.raspberrypi.org.

Używanie małych komputerów do tworzenia projektów, które wykonują i wyczuwają rzeczy

Wcześniej w tej części mówiliśmy o komputerach w kuchni. Wszystkie te komputery wyczuwają otoczenie (na przykład temperaturę piekarnika) i większość z nich robi coś, co ma wpływ na środowisko (na przykład blender sieka lód na fajną Margaritę). Ten puls, puls, puls twojego blendera jest kontrolowany przez komputer. Abyśmy mogli budować projekty i uczyć się, jak projektować własne (a wiercie mi, po zapoznaniu się ze sprzętem będziecie mogli projektować magiczne rzeczy) musimy po prostu wskoczyć i wykonać nasz pierwszy projekt. Następnie, w kolejnych rozdziałach, zbudujemy bardziej złożone rzeczy, które będą punktem wyjścia do Twoich własnych projektów, wszystko zaprogramowane w Pythonie! Ostatnia uwaga, zanim przejdziemy dalej. Oprogramowanie, które będziesz pisać w Pythonie, jest kluczem do uruchomienia wszystkich tych projektów i komputerów. Ważniejszy jest sprzęt czy oprogramowanie? To święta wojna między inżynierami, ale naprawdę uważamy, że oprogramowanie jest ważniejszą częścią i łatwiejszą do nauczenia się dla początkujących. Teraz, zanim to stwierdzenie wyzwoli setki paskudnych e-maili, pozwólcie mi pokornie przyznać, że żadne, i mamy na myśli żadne, z tego nie byłoby możliwe, gdyby nie sprzęt i znakomici inżynierowie, którzy tworzą te małe cuda. Ale to jest książka o Pythonie!

JAKIE SĄ NIEKTÓRE INNE DOSTĘPNE MAŁE KOMPUTERY?

Istnieją dosłownie setki różnych typów małych komputerów i płyt głównych do tworzenia projektów. Wybraliśmy Raspberry Pi ze względu na łatwość obsługi Pythona i setki bibliotek Pythona dostępnych dla Raspberry Pi. Jest to również najlepiej obsługiwany mały komputer z setkami stron internetowych (w tym wspaniałym www.raspberrypi.org), które uczą, jak skonfigurować i używać tego komputera. W rzeczywistości istnieją dwie ogólne kategorie małych systemów komputerowych, które są dostępne dla początkującego użytkownika. Istnieją komputery oparte na systemie operacyjnym Linux (Raspbian, oprogramowanie na Raspberry Pi, jest formą Linuksa) i komputery, które mają znacznie mniejszy system operacyjny lub nawet nie mają systemu operacyjnego. Zrozum, że obie wersje komputerów i systemów operacyjnych są bardzo przydatne w różnych aplikacjach. Raspberry Pi korzysta z systemu Linux. Linux to wielozadaniowy, złożony system operacyjny, który może działać z wieloma rdzeniami procesora. (Czy wspominaliśmy, że Raspberry Pi 3B ma cztery procesory na chipie? A za 35 USD — niesamowite!) Nie należy jednak mylić złożoności systemu operacyjnego z możliwością korzystania z komputera. System operacyjny Raspberry Pi obsługuje cały GUI (graficzny interfejs użytkownika) podobny do systemu Windows, podobnie jak komputer PC lub Mac. Moc systemu operacyjnego to umożliwia. Arduino to małe komputery, które mają tylko mały komputer i ograniczoną ilość pamięci RAM na pokładzie. Co ciekawe, mimo że są znacznie mniejsze i prostsze niż Raspberry Pi, płytki rozwojowe są mniej więcej w tej samej cenie. Jednak pod względem objętości komputer typu Arduino jest znacznie, znacznie tańszy niż Raspberry Pi. Arduino ma znacznie więcej pinów wejścia-wyjścia niż Raspberry Pi i ma wbudowany ADC (analogowy konwerter cyfrowy), coś, czego brakuje Raspberry Pi. W dalszej części pokażemy, jak stworzyć projekt z zewnętrznym ADC i Raspberry Pi. I wiąże się to z płomieniem. Wiesz, że to będzie zabawne. Inną klasą małych komputerów podobnych do Arduino (które mogą być programowane przez to samo IDE (zintegrowane środowisko programistyczne), z

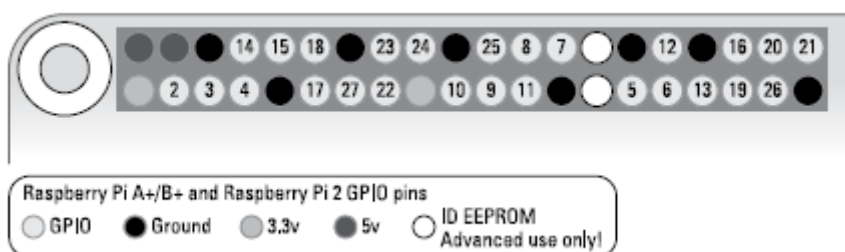
których korzystają urządzenia Arduino, są płytki ESP8266 i ESP32 firmy Espressif w Chinach. Małe komputery mają znacznie mniej pamięci RAM niż Raspberry Pi, ale mają wbudowane interfejsy Wi-Fi (a czasami Bluetooth), dzięki czemu są przydatne w budowaniu projektów, które chcesz połączyć z Internetem, takich jak projekty IOT (Internet Of Things). Obydwa typy komputerów są fajne do zabawy, ale Raspberry Pi ma znacznie lepsze środowisko do programowania i nauki Pythona.

Raspberry Pi: doskonała platforma do obliczeń fizycznych w Pythonie

Do tej pory masz skonfigurowany i działający komputer Raspberry Pi na monitorze, klawiaturze i myszy. Jeśli nie, zrób to teraz (pamiętaj o naszym przyjacielu, www.raspberrypi.org). Kilka następných akapitów będzie o wiele przyjemniejszych z komputerem do pracy! Raspberry Pi to idealna platforma do fizycznych obliczeń w Pythonie, ponieważ ma środowisko wieloekranowe, dużo pamięci RAM i pamięci masowej do zabawy oraz wszystkie narzędzia do tworzenia projektów, które chcemy. W tym rozdziale dużo mówiliśmy o komputerach, a niewiele o Pythonie. Czas to zmienić. Ogromną i potężną cechą Raspberry Pi jest rząd pinów GPIO (wejście-wyjście ogólnego przeznaczenia) wzdłuż górnej części Raspberry Pi. Jest to 40-pinowe złącze, do którego możemy podłączyć dużą liczbę czujników i kontrolerów, aby robić niesamowite rzeczy, rozszerzając Raspberry Pi.

Piny GPIO

Piny GPIO można wyznaczyć (używając oprogramowania Python) jako piny wejściowe lub wyjściowe i używać do wielu różnych celów. Istnieją dwa piny zasilania 5 V, dwa piny zasilania 3,3 V i pewna liczba pinów uziemienia, które mają ustalone zastosowania (zobacz opis napięć (V) w następnym rozdziale i różnice między 3,3 V a 5 V). Pin wyjściowy pinu GPIO „wyprowadza” 1 lub 0 z komputera na pin. Zobacz następną część, aby dowiedzieć się więcej o tym, jak to się robi i co to oznacza. Zasadniczo „1” to 3,3 V, a „0” to 0 V. Możemy myśleć o nich jak o jedynkach i zerach.



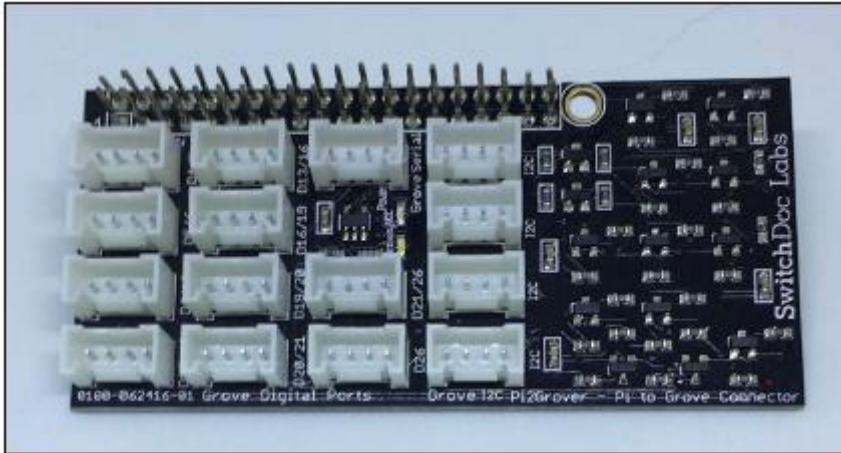
Biblioteki GPIO

Istnieje wiele bibliotek GPIO Python, których można używać do budowania projektów. Ta, której używamy w pozostałej części tej książki, to biblioteka gpiozero, która jest instalowana we wszystkich wersjach oprogramowania komputerowego Raspberry Pi. Dokumentacja biblioteki i instrukcje instalacji (w razie potrzeby) znajdują się na <https://gpiozero.readthedocs.io/en/stable/>. Teraz przejdziemy do fizycznego projektu obliczeniowego „Hello World” z o

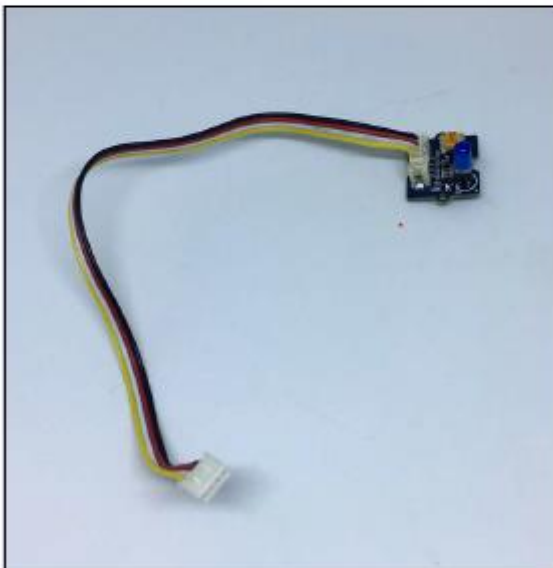
Sprzęt do „Hello World”

Aby wykonać ten projekt, potrzebujemy trochę sprzętu. Ponieważ w pozostałej części książki używamy złączy Grove, przygotujmy dwa elementy sprzętu Grove potrzebne do tego projektu:

Pi2Grover: Konwertuje złącze pinowe Raspberry Pi GPIO na złącza Grove (łatwość użycia i brak możliwości odwrócenia styków zasilania!). Możesz to kupić na shop.switchdoc.com lub na Amazon.com. Możesz otrzymać 5,00 \$ zniżki na płytę Pi2Grover na stronie shop.switchdoc.com, korzystając z kodu rabatowego PI2DUMMIES na stronie checkout.ur Raspberry Pi.



Grove blue LED: moduł Grove blue LED wraz z kablem Grove. Możesz to kupić na shop.switchdoc.com lub na amazon.com.



Montaż sprzętu

Dla wielu czytelników będzie to pierwszy raz, kiedy zmontujesz fizyczny produkt oparty na komputerze. z tego powodu przedstawimy ci proces krok po kroku:

1. Zidentyfikuj płytę Pi2Grover z rysunku powyżej.

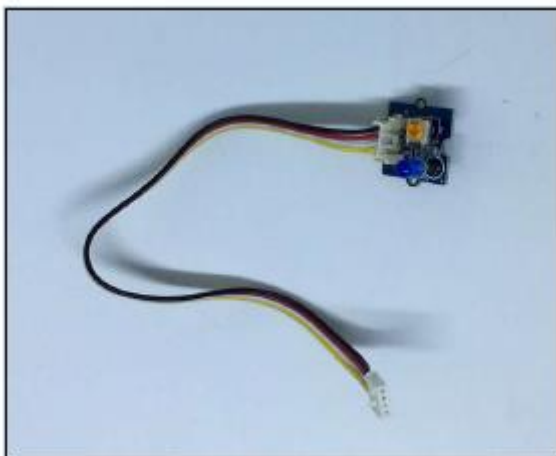
2. Upewniając się, że styki są odpowiednio wyrównane, delikatnie dociśnij płytę Pi2Grover (część A) do 40-pinowego złącza GPIO w Raspberry Pi.



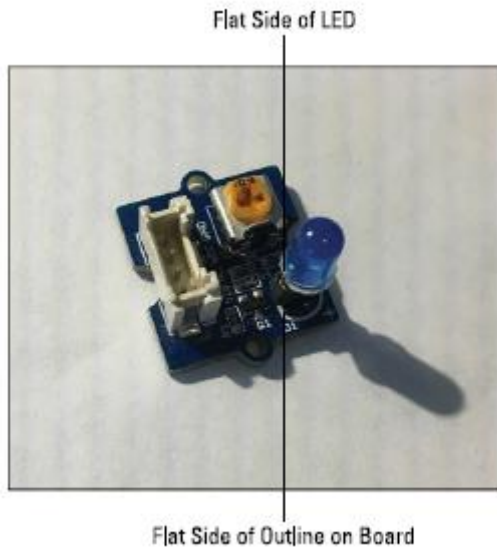
3. Delikatnie dociśnij Pi2Grover (część A) do pinów Raspberry Pi GPIO, upewniając się, że piny są wyrównane. Na obu końcach nie będą widoczne żadne styki i upewnij się, że żadne styki na Raspberry Pi nie są wygięte.



4. Podłącz jeden koniec kabla Grove do niebieskiej płytki LED Grove.

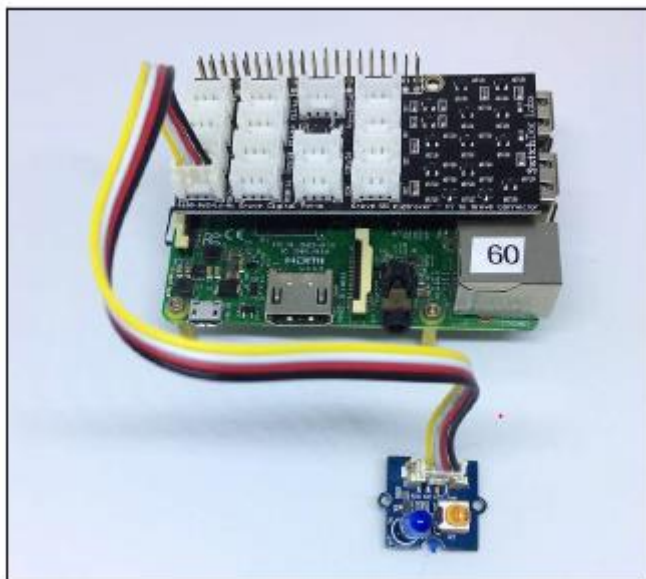


5. Jeśli niebieska dioda LED nie jest podłączona do płytki Grove blue LED, podłącz diodę LED tak, aby lewa strona była wyrównana z lewą stroną konturu na płycie



6. Podłącz drugi koniec kabła Grove do gniazda oznaczonego jako D12/D13 na płycie Pi2Grover.

Teraz zakończyłeś składanie sprzętu. Teraz czas na oprogramowanie Python.



Sterowanie diodą LED za pomocą Pythona na Raspberry Pi

Teraz, gdy mamy już podłączony sprzęt, możemy podłączyć zasilanie do Raspberry Pi. Jeśli wszystko jest w porządku, zobaczysz niebieską diodę LED Grove, niebieską diodę LED zasilania na płycie Pi2Grover, migającą żółtą diodę LED (przez chwilę podczas uruchamiania) na Raspberry Pi i stałą czerwoną diodę LED również na RaspberryPi. Niebieska dioda LED Grove zapala się, gdy włączamy zasilanie Raspberry Pi, ponieważ piny GPIO na Raspberry Pi włączają się jako wejścia. Ponieważ jest to wejście i nic nie steruje pinem GPIO (dioda Grove LED chce wyjścia, a nie wejścia do sterowania diodą LED), pin GPIO po prostu unosi się (nazywany technicznie byciem w trzech stanach). Ze względu na

obwody na płycie Pi2Grover, wejście przesunie się w kierunku „1”, a więc dioda LED się włączy. Kiedy zmienisz pin GPIO na wyjście w poniższym kodzie, dioda LED zgaśnie. Aby rozpocząć, wykonaj następujące kroki:

1. Podejdź do klawiatury i otwórz okno terminala. Jeśli nie wiesz, jak otworzyć i używać okna terminala i wiersza poleceń na Raspberry Pi, przejdź do

<https://www.raspberrypi.org/documentation/usage/terminal/>, aby uzyskać doskonały samouczek.

2. Wprowadź następujący kod Pythona do pliku za pomocą edytora tekstu nano lub innego wybranego edytora. Zapisz go w pliku HelloWorld.py.

```
from gpiozero import LED
```

```
from time import sleep
```

```
blue = LED(12)
```

```
while True:
```

```
    blue.on()
```

```
    print( "LED On")
```

```
    sleep(1)
```

```
    blue.off()
```

```
    print( "LED Off")
```

```
    sleep(1)
```

Aby uzyskać doskonały samouczek dotyczący korzystania z nano, przejdź do <https://www.raspberrypi.org/magpi/edit-text/>

3. Teraz wielki moment. Uruchom program, uruchamiając to w wierszu poleceń w oknie terminala:

```
sudo python3 HelloWorld.py
```

Dioda LED będzie migać raz na sekundę, a na ekranie w oknie terminala pojawi się następujący komunikat:

```
LED On
```

```
LED Off
```

```
LED On
```

```
LED Off
```

```
LED On
```

```
LED Off
```

```
LED On
```

```
LED Off
```

```
LED On
```


LED Off

LED On

Słowo kluczowe sudo oznacza superużytkownik do. W tego typu kodzie używamy sudo przed poleceniem python3, ponieważ niektóre wersje systemu operacyjnego Raspberry Pi ograniczają dostęp zwykłego użytkownika do niektórych pinów i funkcji. Używając Sudo, uruchamiamy to jako superużytkownik. Oznacza to, że będzie działać bez względu na konfigurację konkretnej wersji systemu operacyjnego. W nowszych wersjach systemu operacyjnego Raspberry Pi wystarczy wpisać python3 HelloWorld.py i zadziała. Jeśli nie, wróć do sudo python3 HelloWorld.py. Możesz zatrzymać ten program za pomocą Ctrl C (^C, w kategoriach maniaków). W kodzie poniższa instrukcja importuje funkcję LED z biblioteki Pythona gpiozero:

```
from gpiozero import LED
```

Ta instrukcja importuje funkcję sleep z biblioteki czasu Pythona:

```
from time import sleep
```

To przypisuje diodę LED na GPIO 12 (pamiętasz D12/D13 na płycie Pi2Grover?):

```
blue = LED(12)
```

Teraz zaczynasz pętlę, która będzie trwać wiecznie:

```
while True:
```

Włącz diodę:

```
blue.on()
```

```
print("LED On")
```

Poczekaj, aż minie jedna (1) sekunda:

```
sleep(1)
```

Wyłącz diodę LED:

```
blue.off()
```

```
print("LED Off")
```

```
sleep(1)
```

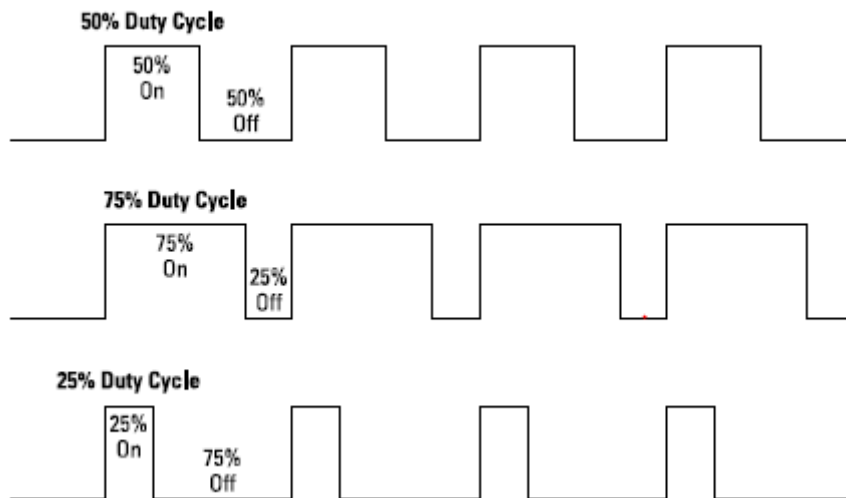
Wyptukać i powtórzyć.

Wow, właśnie wkroczyłeś do świata obliczeń fizycznych. Poczekaj tylko, aż skończysz tę książkę. Będziesz zdumiony, co możesz zrobić!

Ale poczekaj, jest więcej. . .

Ponieważ mamy już skonfigurowany cały ten sprzęt, może zrobimy jeszcze jeden interesujący projekt? Zrobimy z tego diodę LED o zmiennej jasności, używając PWM (modulacja szerokości impulsu), aby zmieniać jasność diody LED. Modulacja szerokości impulsu (PWM) to technika, za pomocą której zmieniasz czas, przez jaki sygnał jest na poziomie 1, w porównaniu do czasu, w którym sygnał jest na poziomie 0. Ponieważ nasza dioda LED włącza się, gdy jest na poziomie 1 i wyłącza się przy 0, jeśli

zmienimy czas, w którym jest 1 w stosunku do 0, możemy kontrolować jasność ludzkiego oka. Ten stosunek nazywa się cyklem pracy.



100-procentowy cykl pracy oznacza, że jest włączony przez 100 procent czasu, podczas gdy cykl pracy 0 procent oznacza, że jest wyłączony przez cały czas. Zmiana czasu włączenia sygnału spowoduje zmianę jasności diody LED.

Wprowadź ten kod Pythona do nano i zapisz go jako HelloWorld2.py:

```
from gpiozero import PWMLED
from time import sleep
led = PWMLED(12)
while True:
    led.value = 0 # off
    sleep(1)
    led.value = 0.5 # half brightness
    sleep(1)
    led.value = 1 # full brightness
    sleep(1)
```

Teraz uruchom kod:

```
sudo python3 HelloWorld2.py
```

Zobaczysz zmianę jasności co sekundę. I jeszcze jedno, oto jak zmienić jasność w sposób ciągły:

```
from gpiozero import PWMLED
from signal import pause
led = PWMLED(12)
```

```
led.pulse()
```

```
pause()
```

Za pomocą tego kodu widzimy płynne ciągłe rozjaśnianie i ściemnianie diody LED. Chłopcze, dużo osiągnąłeś w tej części. Teraz zacząłeś dostrzegać możliwości fizycznych obliczeń. I masz niebieską diodę!

ZMIANA DIOD LED NIE JEST CAŁKOWICIE PŁYNNĄ

Okazuje się, że sposób, w jaki działa system operacyjny Raspberry Pi Linux, powoduje, że Twój program nie jest jedyną rzeczą działającą w tym samym czasie. Jeśli chcesz zobaczyć wszystko, co działa na twoim Raspberry Pi, wpisz `ps xaf` w wierszu poleceń na swoim terminalu. Będziesz zaskoczony tym, co działa na twoim Raspberry Pi. Ponieważ system operacyjny na Raspberry Pi jest wielozadaniowy, co oznacza, że jednocześnie działa więcej niż jedno zadanie, czasami twoje zadanie PWM (tak jak jest uruchamiane w oprogramowaniu) nie obciąża procesora dokładnie wtedy, kiedy tego chce i dlatego jest tylko trochę drgań diody LED. Raspberry Pi ma dwa sprzętowe piny PWM GPIO, których można użyć, jeśli nie używasz wyjścia audio w Raspberry Pi. Na Raspberry Pi 3B ledwo to zauważysz, ale na wolniejszych wersjach Pi.