

5. ZAPROGRAMUJ SYSTEM MONITOROWANIA I STEROWANIA SZKLARNIĄ

Co mają ze sobą wspólnego zwykły programowalny termostat w domu i tempomat w samochodzie? Oba urządzenia pozwalają użytkownikowi ustawić żadaną wartość wielkości, która jest daną wejściową dla programu i utrzymywać ją w pobliżu tej wartości. Jednak program tempomatu jest bardziej złożony niż program termostatu, ponieważ bardzo ważne jest, aby prędkość pojazdu pozostawała bardzo blisko wybranej wartości.

Standardowy program termostatu opiera się na *kontroli typu bang-bang*. Gdy jest zimno, a temperatura spadnie do wartości o kilka stopni poniżej ustawionej, *bang!* – grzejnik włącza się. Grzejnik działa, aż temperatura wzrośnie o kilka stopni powyżej idealnej, a następnie *bang!* – grzejnik wyłącza się. Jeśli wykonałeś którekolwiek z poprzednich doświadczeń dotyczących Szklarni PASCO, zaprogramowałeś już system sterowania *typu bang-bang*, w którym wartość jednej lub większej liczby danych wejściowych określała, czy urządzenie wyjściowe powinno być włączone, czy wyłączone. *Sterowanie proporcjonalne* to tylko jeden z elementów strategii programowania, którą wykorzystuje tempomat do utrzymania stałej prędkości pojazdu. Wymaga ono również danych wejściowych pochodzących z czujników, ale urządzenia wyjściowe są *regulowane* lub ustawiane tak, aby stale dostosowywały się do sytuacji poprzez zwiększanie lub zmniejszanie swojej mocy, a nie włączanie lub wyłączanie się. Prędkość pojazdu jest ściśle monitorowana, a system stale dostosowuje takie ustawienia, jak położenie przepustnicy, aby jak najbardziej zbliżyć się do idealnej prędkości. Podczas tych zajęć napiszesz 24-godzinny program, który połączy dowolną kombinację nowego, zmodyfikowanego i istniejącego kodu z co najmniej jednym zastosowaniem *sterowania proporcjonalnego*, aby żywa roślina miała optymalne warunki i zdrowo rozwijała się w całkowicie zautomatyzowanej szklarni.

Cele

- Wykorzystanie istniejącego kodu do budowy nowego programu.
- Zaprojektowanie programu łączącego różne mechanizmy sterowania w celu automatycznego zarządzania szklarnią.

Materiały i wyposażenie

- System gromadzenia danych, SPARKvue
- `//control.Node`
- *Lampa do wzrostu roślin* z dołączonymi kablami i zasilaczem USB
- *Czujnik parametrów w szklarni*
- Moduł sensorów *czujnika parametrów w szklarni* z przewodem i korkiem
- Sonda wilgotności gleby
- *Wentylator USB*
- *Płytką zasilającą* z przewodem
- *Eko-komora* z pokrywą i korkami
- Zmontowany system nawadniania*
- Roślina zasadzona w doniczce o średnicy ~10 cm i wysokości ~10 cm
- Płytkie naczynie
- Plastikowa torba z suwakiem (strunowa) o wymiarach ~ 17 cm x 20 cm
- Lód

*Zawiera wszystkie elementy zestawu akcesoriów do szklarni (rurki, złącza, dysze nawadniające, zapięcia na rzepy i korek #5 z jednym otworem), zbiornik wypełniony wodą z kranu do zasilania pompy USB oraz materiały do mocowania rurek do doniczki, takie jak kilka mocnych gumek, opasek zaciskowych i klipsów biurowych. Systemy, które wymagają zwiększonej wilgotności, mogą obejmować gąbkę o wymiarach 5 cm x 5 cm i nakrętki do butelek, jednak po umieszczeniu rośliny w szklarni konfiguracja może zostać zmodyfikowana, gdy zmienia się warunki wilgotności.

Bezpieczeństwo

Postępuj zgodnie z regulaminem swojej pracowni, a ponadto:




- trzymaj wodę z dala od czujników, wtyczek elektrycznych i odsłoniętych płytek elektronicznych.
- nie dopuść, aby odsłonięte płytki elektroniczne stykały się z metalową lub przewodzącą powierzchnią.

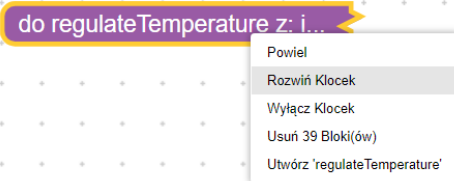
OSTROŻNIE:

- nie patrz bezpośrednio na diody LED.
- nie dotykaj diod LED.

Wstęp

Ta część pomoże ci zrozumieć, jak skonfigurować i używać programowania do sterowania proporcjonalnego. Wykonaj następujące czynności:

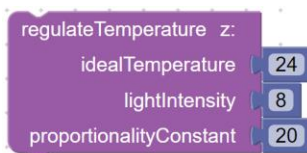
1. Podłącz `//control.Node` ze swoim urządzeniem, wybierz dowolny szablon, aby utworzyć ekran wyświetlania danych, a następnie otwórz narzędzie **Kodowanie** .
2. Otwórz **bibliotekę kodów PASCO**  w prawym górnym rogu.
3. Zmień kategorię z *PASCObot* na *Szklarnia* i wybierz *Regulacja temperatury*.
4. Kliknij prawym przyciskiem myszy lub naciśnij i przytrzymaj blok *do regulateTemperature z:* aby otworzyć menu, jak pokazano, a następnie wybierz opcję **Rozwiń Klocek**.
5. Przeczytaj po angielsku dymek komentarza , aby zorientować się, do czego służy ta funkcja.
6. Przeanalizuj budowę i organizację kodu. Poszukaj połączeń między edytowalnymi wartościami wejściowymi funkcji w górnym bloku (wstępnie ustawionymi na 24, 8 i 20) a samym kodem w rozwiniętym bloku funkcji. Zanim zaczniesz czytać dalej, zastanów się, jak działa ten proporcjonalny system sterowania (*Wskazówka: element w porcie B, CH1 to wentylator USB*).



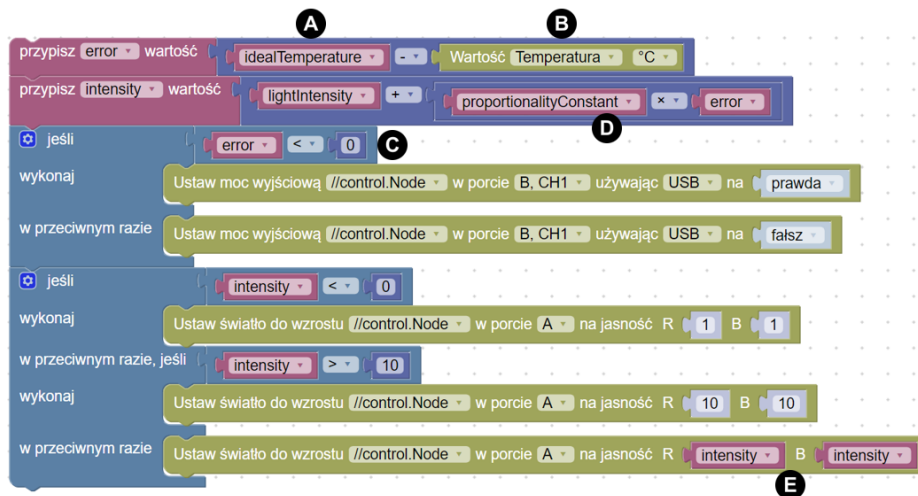
Wentylator USB wykorzystuje się do sterowania *bang-bang*, aby utrzymać temperaturę szklarni w pobliżu *wartości idealnej*, gdy włączona jest *Lampa do wzrostu roślin*. Jeśli sam wentylator nie jest w stanie utrzymać *idealnej temperatury*, *lightIntensity* (jasność światła) zmniejsza się poniżej wartości idealnej, aż temperatura ustabilizuje się w pobliżu tej, którą chcemy uzyskać. Gdy temperatura jest już ustabilizowana, natężenie światła może stopniowo wzrastać, pod warunkiem, że temperatura jest właściwa. Ale jaka jest rola zmiennych *error* i *proportionalityConstant*?

Najpierw przyjrzyjmy się 3 edytowalnym wejściom funkcji:

- *idealTemperature* – temperatura docelowa, jest ustawiona na 24 °C.
- *lightIntensity* – jasność czerwonego i niebieskiego światła *Lampy do wzrostu roślin*, początkowo ustawiona na poziom 8.
- *proportionalityConstant* – ma wartość 20, ale ... co to właściwie oznacza?



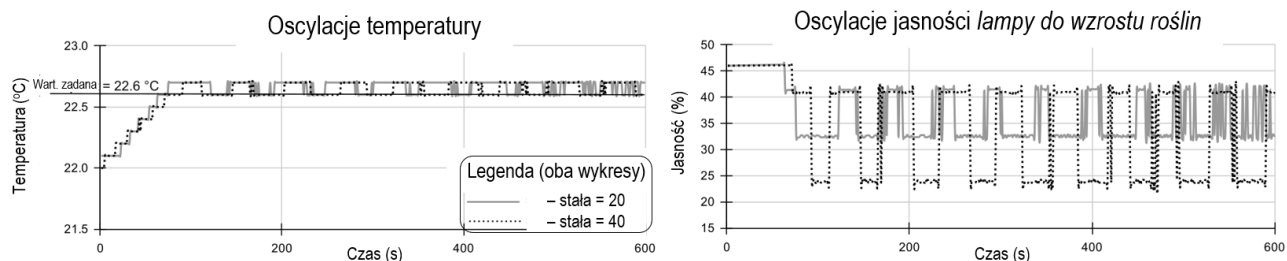
Jak widać w kodzie, programy sterowania proporcjonalnego wymagają 3 elementów: „idealnej” wartości zmiennej zwanej *wartością zadaną* **A**; pomiaru z czujnika zwanego *zmienną procesową* **B**, która służy porównaniu danych w czasie rzeczywistym do wartości zadanej; oraz *błędu* (zwanego także *uchyblem*), który jest różnicą między wartością zadaną a rzeczywistą, **A** – **B**.



Im bardziej rzeczywista temperatura odbiega od wartości zadanej, tym większy jest błąd. Jeśli nie ma różnicy między rzeczywistą temperaturą a wartością zadaną, błąd wynosi zero, a system jest stabilny. Jeśli zmierzona temperatura jest wyższa od wartości zadanej, błąd jest ujemny, czyli poniżej zera **C** i wentylator zostanie włączony, w przeciwnym razie zostanie wyłączony.

Wentylatora USB nie można ustawić na różne poziomy mocy, więc można używać tylko sterowania bang-bang. *Lampa do wzrostu roślin* ma regulowaną intensywność, dzięki czemu można dla niej zastosować sterowanie proporcjonalne. Wydajność lampy jest *proporcjonalna* do błędu, co oznacza, że stosunek natężenia światła do błędu jest wartością stałą, równą *stałej proporcjonalności*. Aby obliczyć zmienną *intensity*, mnoży się tę stałą przez błąd **D** i dodaje do jasności początkowej, określonej w danych wejściowych funkcji. Zmienna *intensity* jest następnie używana do aktualizacji pracy *lampy do wzrostu roślin* **E**, jeśli zajdzie taka potrzeba.

Wielkość błędu i jego znak dodatni lub ujemny mają znaczenie. Błąd ujemny powoduje ustawienie intensywności niższej niż początkowa, podczas gdy błąd dodatni (gdy rzeczywista temperatura jest niższa niż wartość zadana) powoduje ustawienie intensywności wyższej niż początkowa. Wyższe wartości stałych proporcjonalności powodują większe zmiany w natężeniu światła, gdy temperatura w szklarni oddala się od idealnej.



Powyzsze wykresy pokazują dane dotyczące temperatury (po lewej) i jasności (po prawej) dla dwóch 10-minutowych serii pomiarowych z funkcją *regulateTemperature* w pętli *powtarzaj dopóki prawda*. Wartość zadana wynosiła 22,6 °C dla obu serii. Linia ciągła pokazuje wyniki, gdy stała proporcjonalności była ustawiona na 20, a linia kropkowana – dla stałej ustawionej na 40. Po lewej

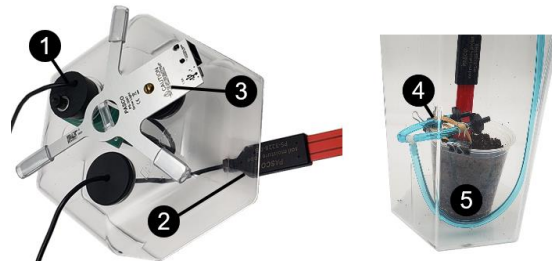
stronie obie stałe powodują, że temperatura oscyluje (wielokrotnie wzrasta i spada) wokół wartości zadanej. Czy zauważyłeś, że gdy stała jest ustawiona na 40, temperatura jest bardziej stabilna między oscylacjami? Teraz spójrz na różnicę w oscylacjach intensywności światła po prawej stronie. Jasność spada *znacznie bardziej*, gdy stała ma większą wartość, a więc do komory dostaje się mniej intensywne światło i mniej ciepła – dlatego temperatura jest bardziej stabilna przy wyższej stałej proporcjonalności.

Prototyp

Część 1: Konfiguracja

Konfiguracja szklarni

1. Wyłącz `//control.Node` i zamknij SPARKvue.
2. W razie potrzeby podlej roślinę.
3. Umieść płytkę z sensorami **1** i sondę wilgotności gleby **2** czujnika parametrów w szklarni oraz lampę do wzrostu roślin **3** w pokrywie szklarni, jak pokazano na rysunku 1.
4. Delikatnie przeciągnij około 30 cm przewodu sondy wilgotności gleby przez pokrywę.
5. Użyj spinaczy i gumek, **4** aby przymocować rurki i dysze nawadniające do doniczki, jak pokazano na rysunku 1 (w doniczce powinna znajdować się roślina).
6. Wsuń sondę wilgotności gleby do doniczki tak daleko, jak to możliwe, w pobliżu dyszy nawadniającej, jak pokazano na rysunku. Ubij glebę wokół sondy.
7. Umieść roślinę w szklarni, dysze nawadniające skieruj lekko do góry, a rurki ułóż w kształcie haczyka jak na rysunku **5**, aby uniknąć pęcherzyków powietrza i nadmiernego drenażu,
8. Opcjonalnie: Jeśli stojąca woda na dnie jest częścią Twojej strategii kontroli wilgotności, rozmieść nakrętki od butelek, dysze i stojącą wodę, zgodnie z potrzebami. Jeśli mokra gąbka jest częścią Twojej strategii, zaprojektuj system utrzymywania gąbki na odpowiedniej wysokości i w odpowiedniej odległości od wentylatora po jego zainstalowaniu.
9. Umieść pokrywę na szklarni i zamocuj korek rurki systemu wodnego i wentylator USB w bocznych otworach.
10. Delikatnie przeciągnij nadmiar przewodu sondy wilgotności gleby z powrotem przez pokrywę i wyrównaj lampę do wzrostu roślin z czujnikiem światła.
11. Zbuduj system chłodzenia, jak pokazano na rysunku 2. Włóż kostki lodu do płytkiego naczynia tak, by było prawie pełne **A**.



Rysunek 1: Konfiguracja szklarni

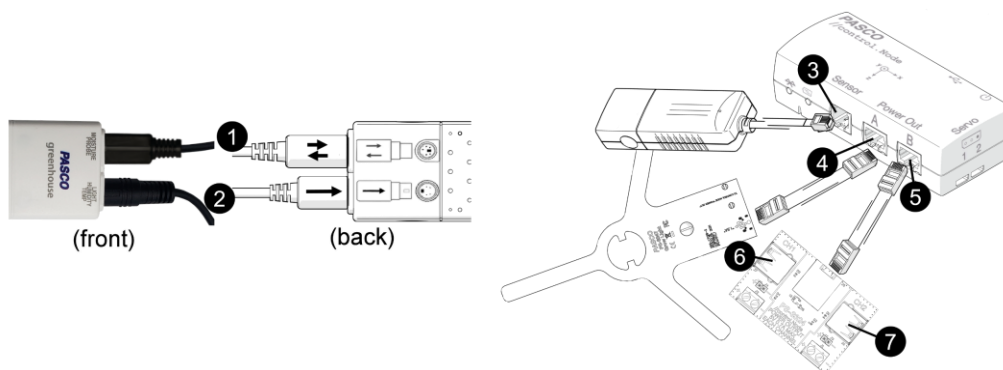


Rysunek 2: Układ chłodzenia umieszczony pod pokrywą

- Umieść naczynie w woreczku strunowym, usuń nadmiar powietrza z worka i zamknij go.
- Systemu chłodzenia należy użyć, gdy konieczne jest szybkie obniżenie temperatury podczas testowania. W tym celu należy ustawić pokrywę komory na stole z układem chłodzenia umieszczonym bezpośrednio pod korkiem, w którym znajduje się czujnik temperatury na płytce sensorów (patrz: strzałka **B**). Nie wolno dopuścić do kontaktu układu chłodzenia z jakąkolwiek częścią modułu czujnika.

Konfiguracja sprzętu

- Podłącz sondę wilgoci gleby **1** oraz płytkę z sondami światła, wilgotności powietrza i temperatury **2** do czujnika parametrów w szklarni, jak pokazano na rysunku 3. Podłącz czujnik do //control.Node **3**.






Rysunek 3: Konfiguracja sprzętu


- Podłącz lampę do wzrostu roślin do portu A //control.Node **4**. Podłącz lampę do zasilacza USB.
- Podłącz płytkę zasilającą do portu B //control.Node **5**.
- Podłącz wentylator USB do kanału 1 płytki zasilającej **6**, a pompę USB do kanału 2 **7**.


Część 2: Przesyłanie kodu do //control.Node

//control.Node ma funkcję, która umożliwia przechowywanie kodu Blockly bezpośrednio w samym urządzeniu — więc nie będzie nawet potrzeby włączania SPARKvue, aby uruchomić program! Może to być przydatne w przypadku szklarni, która musi działać samodzielnie przez wiele dni, tygodni lub miesięcy.





- Otwórz program SPARKvue i połącz //control.Node do swojego urządzenia.
- Wybierz **Temperature** i dowolne inne pomiary czujnika parametrów w szklarni.
- Wyłącz pomiary czujników wbudowanych w //control.Node, zmieniając suwak z włączonego na wyłączony .
- Wybierz dowolny szablon wyświetlania danych. Gdy otworzy się ekran, rozpocznij zbieranie danych, aby zarejestrować aktualną temperaturę, a następnie zatrzymaj zbieranie danych.
- Użyj ikony , aby otworzyć narzędzie **Kodowanie**.
- Przejdź do **biblioteki kodów PASCO** i wstaw funkcję *Regulacja temperatury szklarni*.

7. Zmień wartość zmiennej *idealTemperature* z wartości domyślnej 24 °C na o 0,2 °C wyższą niż właśnie zarejestrowana temperatura.
8. Umieść blok funkcji wewnątrz pętli *powtarzaj dopóki prawda*. Przetestuj kod przez kilka sekund, aby upewnić się, że działa. W razie potrzeby popraw go.
9. Użyj ikony **Załaduj kod**  w prawym górnym rogu ekranu, aby wysłać kod do `//control.Node`. Usłyszysz serię sygnałów dźwiękowych wskazujących, że przesyłanie kodu powiodło się.
10. Użyj zielonego przycisku **Rozpocznij wykonywanie kodu**  w prawym górnym rogu, aby rozpocząć uruchamianie programu zapisanego w `//control.Node`. Gdy przekazany kod jest uruchomiony, na `//control.Node` pojawia się migające niebieskie światło.
11. Zamknij SPARKvue. Zauważ, że urządzenia wyjściowe w szklarni nadal działają, a niebieskie światło miga.
12. Otwórz program SPARKvue i połącz `//control.Node`, a następnie powtórz kroki 2 i 3.


UWAGA: Przesłany kod może zwracać tylko pomiary czujnika. Dane tekstowe i numeryczne nie mogą być pobierane z kodu przesłanego do `//control.Node`. Tekstowe i liczbowe elementy wyjściowe można wyświetlać na ekranie tylko wtedy, gdy wykonanie kodu jest inicjowane za pomocą przycisku **Start**  w oprogramowaniu SPARKvue.

13. Zaczynaj zbierać dane; Poczekaj kilka sekund na zarejestrowanie danych, a następnie zatrzymaj zbieranie danych. Zwróć uwagę, że przekazany program nadal działa, niezależnie od tego, czy zbieranie danych jest uruchomione, czy nie.
14. Aby zatrzymać działanie programu w `//control.Node`, otwórz narzędzie **Kodowanie** i wybierz ikonę **Zatrzymaj wykonywanie kodu**  w prawym górnym rogu lub wyłącz `//control.Node`.

Część 3: Przegląd istniejącego kodu

1. Przejdź do menu głównego w SPARKvue  i wybierz pozycję **Otwórz...** Przejdź do wcześniej zapisanej pracy z dowolnego poprzedniego badania szklarni, która zawiera działający kod i otwórz plik. Alternatywnie, możesz skorzystać ze zrzutów ekranu z kodem lub szkiców wykonanych na poprzednich zajęciach.
2. Przejrzyj swój kod. Zdecyduj, gdzie należy wprowadzić zmiany, aby program działał w cyklu 24-godzinny; Rób notatki podczas pracy.
3. Powtórz kroki 1-2 dla wszystkich dotychczasowych zadań kodowania szklarni. W razie potrzeby użyj funkcji z biblioteki kodów.
4. Przebuduj lub otwórz i uruchom dowolny wcześniej utworzony kod, który ma dane wyjściowe tekstowe. Podczas zbierania danych przejdź do ekranu wyświetlania danych i utwórz **Nową stronę** .
5. Wybierz układ całej strony  i dodaj tabelę .
6. Kliknij opcję **Wybierz pomiar**, aby dodać co najmniej jeden rodzaj pomiaru pochodzący z czujnika parametrów w szklarni i co najmniej jedno wyjście tekstowe kodu. Aby dodać

wyjściowe dane tekstowe, wybierz opcję **Wybierz pomiar**, a następnie przełącz się z karty **Czujniki** na kartę **Dane użytkownika** w menu, które otworzy się po prawej stronie.

7. Otwórz menu narzędzi tabel  pod tabelą, w którym można dodać lub usunąć kolumny.
8. Obserwuj dane dodawane do tabeli. Używaj tego typu obserwacji danych podczas rozwiązywania problemów z kodem, ponieważ czasami pomaga to uzyskać nowe informacje.

Część 4: Zaplanuj swój kod

Na osobnym papierze naszkicuj schemat blokowy pokazujący, jak planujesz modyfikować i połączyć istniejące kody w celu utworzenia jednego programu, który:

1. działa w cyklu 24-godzinny, z czasem rozpoczęcia określonym przez użytkownika;
2. wyznacza logiczną kolejność zdarzeń wejściowych i wyjściowych przy użyciu odpowiednich pętli i bloków czasowych;
3. używa funkcji do oddzielania i organizowania zadań programu;
4. zapewnia regularne dostarczanie danych z czujnika (temperatura, wilgotność powietrza, jasność, wilgotność gleby) w celu monitorowania warunków w odpowiednich dla każdego pomiaru odstępach czasu;
5. wykorzystuje wejścia pomiarowe do wyzwalania aktywności pompy USB, wentylatora USB i zmian intensywności *lampy do wzrostu roślin*, które utrzymują preferowaną temperaturę, wilgotność powietrza, jasność i wilgotność gleby dla rośliny przez cały dzień (pamiętaj, aby ustawić „idealną temperaturę” nieco powyżej aktualnej temperatury pokojowej);
6. zawiera alerty wizualne (tekst na wyświetlaczu **Cyfry**) i alerty dźwiękowe (przez głośnik //control.Node);
7. Zawiera sterowanie bang-bang i sterowanie proporcjonalne.

Schemat blokowy powinien zawierać przybliżoną porę dnia, w której występuje każde zdarzenie, a także pomysły na rodzaje bloków, których będziesz używać. Schemat blokowy nie musi pokazywać wszystkich wierszy kodu; jego celem jest pomoc w planowaniu i myśleniu o rzeczach, które muszą się wydarzyć w logicznej kolejności z odpowiednimi rodzajami zmiennych, pętli, opóźnień itp. potrzebnych do prawidłowego działania programu. Na przykład krok schematu blokowego może brzmieć „czekaj godzinę, a następnie użyj pętli *dla każdego elementu* z listą, aby stopniowo włączać światło” zamiast wymieniania listy wielu bloków, które byłyby potrzebne do wykonania tego zadania.

Próba generalna

Rozpocznij nowy eksperyment w programie SPARKvue i zacznij przekształcać schemat blokowy na kod. Pamiętaj, aby testować każdą funkcję kodu podczas pracy, aby łatwiej było rozwiązywać problemy i zapisuj swoją pracę na bieżąco. Dla przypomnienia, na godzinę przypada 3600 sekund, a na 24-godzinny dzień - 86 400 sekund. Pisz kod z myślą o sekundach, ale podczas testowania zmień jednostki czasu z sekund na milisekundy, aby skondensować 1 godzinę w 3,6 sekundy. W ten sposób 24 godziny zostaną zredukowane do zaledwie około 1,5 minuty. Program musi spełniać wszystkie kryteria opisane w części 4 powyżej. Jeśli utkniesz podczas rozwiązywania problemów, spróbuj utworzyć tabelę, aby bezpośrednio zobaczyć wyniki pracy kodu. Zrób zrzut ekranu, a po zakończeniu wydrukuj lub naszkicuj kod na osobnej kartce papieru. Następnie odpowiedz na pytania poniżej. Zapisz swoją pracę w SPARKvue do wykorzystania w przyszłości. Jeśli planujesz pozostawić szklarnię działającą autonomicznie przez dłuższy czas, pozostaw //control.Node podłączony do zasilania USB, załaduj i wystartuj kod na //control.Node przed zamknięciem SPARKvue.

1. Podsumuj zdarzenia występujące w 24-godzinnym programie i wyjaśnij, w jaki sposób program został napisany, aby zapewnić prawidłowy harmonogram w cyklu dobowym.
2. Jakie były 2 największe wyzwania, z którymi musiałeś się zmierzyć podczas łączenia wielu zdarzeń wejściowych i wyjściowych w jednym programie?
3. Jakiej rady udzieliłbyś swoim rówieśnikom na temat tego, jak odnieść sukces, łącząc wiele zdarzeń wejściowych i wyjściowych w jednym programie?
4. Wyjaśnij swoje podejście do wykorzystania sterowania proporcjonalnego w programowaniu cyklu dobowego. Opisz co najmniej jedną zaletę tego typu sterowania w porównaniu ze sterowaniem typu bang-bang.

Ulepszenia

- Zapoznaj się z zaleceniami dotyczącymi nawadniania lub nawożenia Twojej rośliny; dodaj przypomnienia tekstowe i dźwiękowe, aby przestrzegać zalecanego harmonogramu.
- Zamiast sygnałów dźwiękowych zakoduj kilka nut z rozpoznawalnych utworów. Na przykład odtwórz kilka nut podobnych do refrenu z *Umbrella* Rhianny lub *Singin' In The Rain* Gene'a Kelly'ego, które wydobywają się z głośnika podczas gdy pompa USB podlewa rośliny.
- Użyj charakterystycznego, przyciągającego uwagę, zmieniającego kolor lub migającego schematu pracy *lampy do wzrostu roślin*, aby przekształcić dowolny alert tekstowy w alert świetlny.