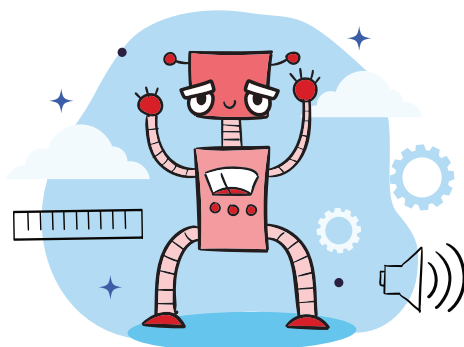




Pomiar odległości i analiza danych pomiarowych



Zagadnienia informatyczne

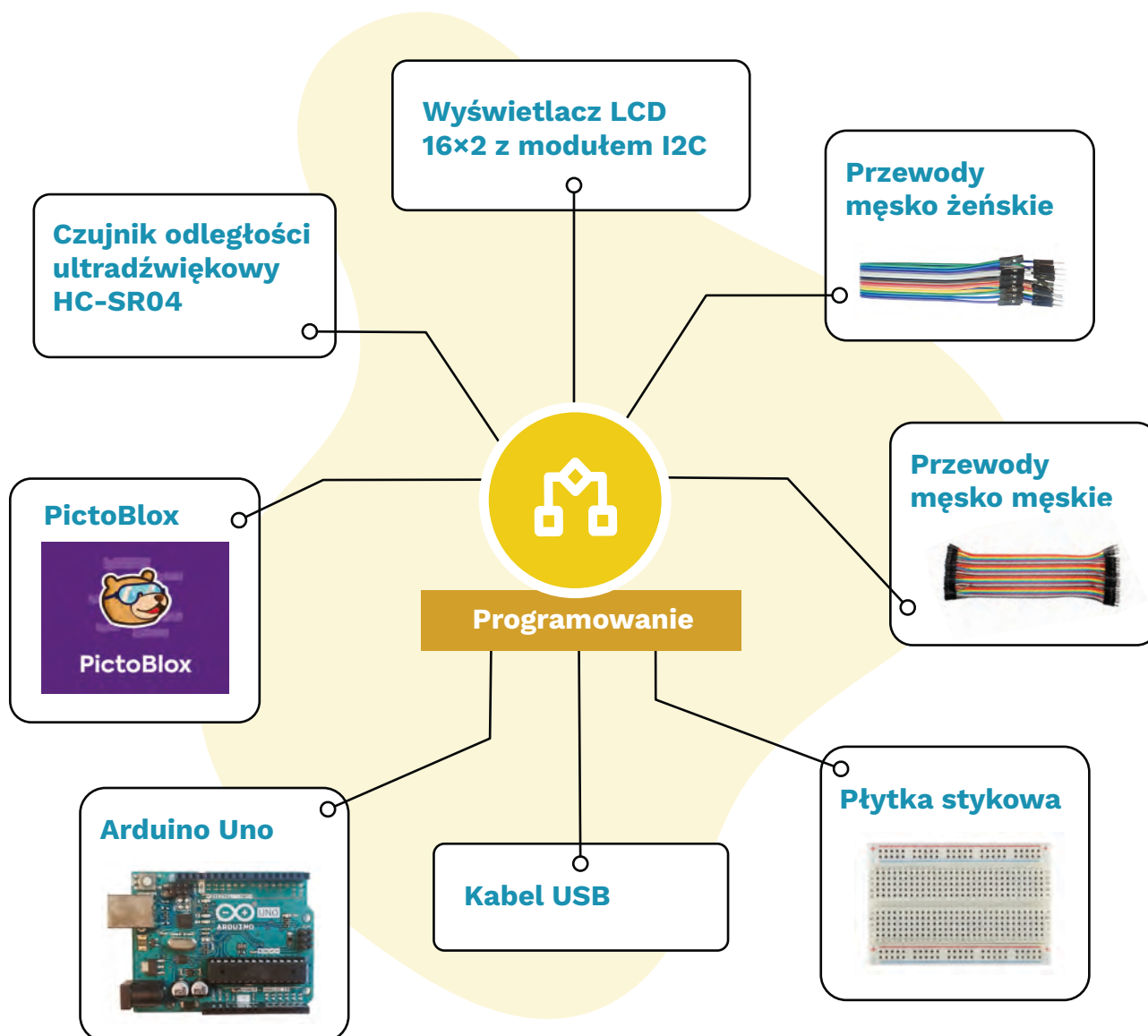
W tej części uczniowie poznają wyświetlacz LCD 16x2 jako interfejs użytkownika oraz uczą się komunikacji z urządzeniami za pomocą magistrali I2C. Dzięki temu mogą wyświetlać tekst, zmienne i wyniki pomiarów bezpośrednio na ekranie podłączonym do Arduino. Następnie wykorzystują czujnik odległości ultradźwiękowy HC-SR04, który dostarcza dane liczbowe w czasie rzeczywistym, oraz tworzą program wyświetlający aktualną odległość i jej interpretację słowną. Uczniowie widzą, w jaki sposób Arduino zbiera dane z otoczenia i prezentuje je w czytelnej formie.



Zagadnienia matematyczne

Część matematyczna koncentruje się na obliczaniu odległości oraz średniej arytmetycznej na podstawie rzeczywistych pomiarów wykonanych przez czujnik ultradźwiękowy. Uczniowie analizują zależność między czasem a odległością oraz wykorzystują wyniki pomiarów do obliczania średniej z kilku wartości. Następnie rozwiązują zadanie odwrotne – wyznaczają brakujący pomiar, który pozwoli uzyskać określoną średnią. Dzięki temu uczą się interpretować dane pomiarowe i rozumieć znaczenie średniej jako narzędzia analizy wyników

Co będzie potrzebne do module 5?



Poznajemy świat Arduino: Wyświetlacz LCD i czujnik odległości – pomiar i prezentacja danych

Wyświetlacza LCD 16x2

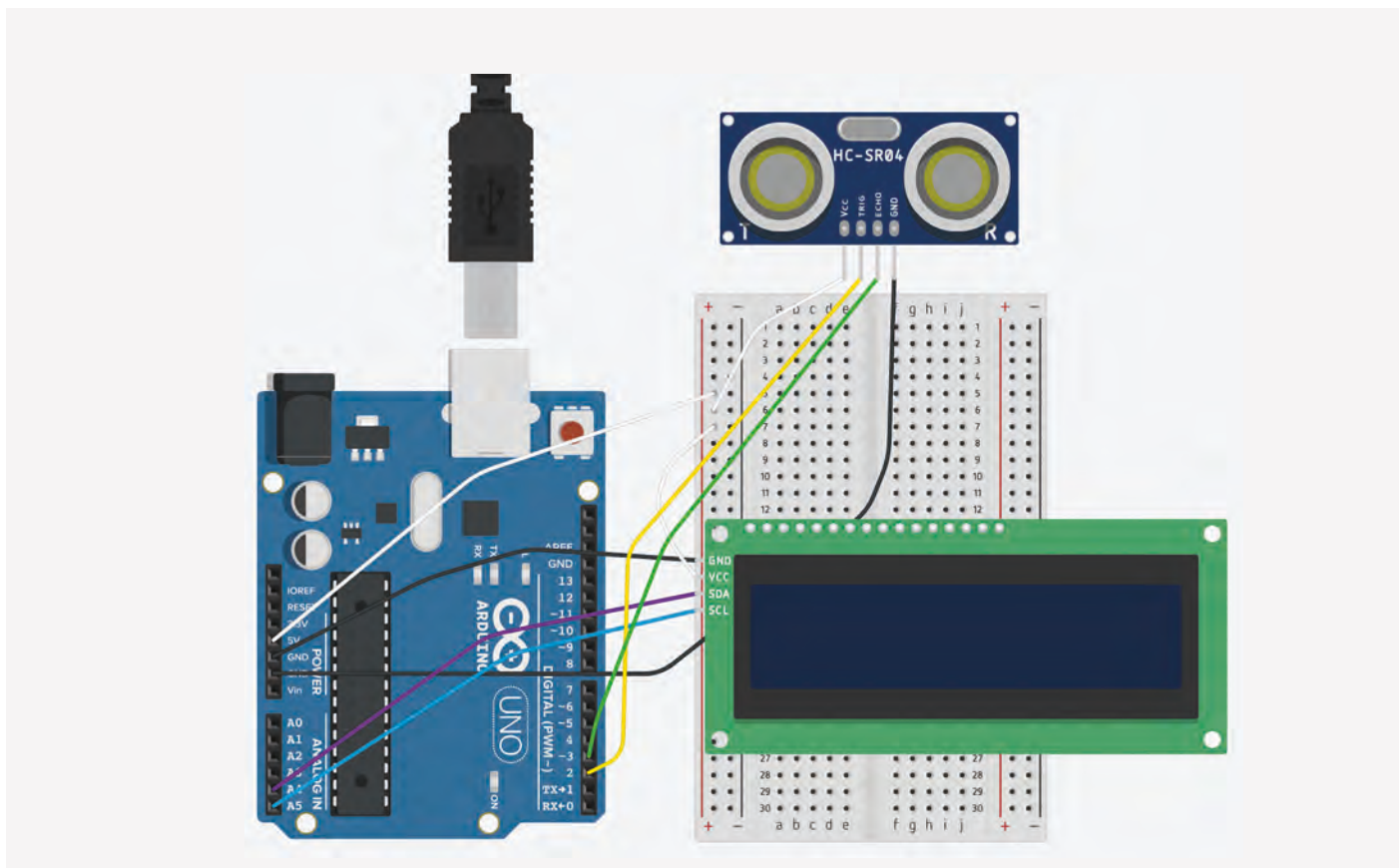
Wyświetlacz LCD 16×2, inaczej LCD 1602 to prosty ekran tekstowy, który potrafi wyświetlić 16 znaków w dwóch wierszach. Jest często spotykany w urządzeniach codziennego użytku, takich jak liczniki, wagi, sterowniki czy panele informacyjne. W projektach z Arduino pełni rolę interfejsu użytkownika – pozwala wyświetlać pomiary, wyniki obliczeń oraz komunikaty bez konieczności korzystania z komputera.

W klasycznej wersji LCD wymaga wielu przewodów do komunikacji z mikrokontrolerem. Dlatego w tym module wykorzystujemy dodatkowy moduł I2C, który jest „doczepiony” do wyświetlacza. Ten mały układ pośredniczący zamienia skomplikowaną komunikację równoległą na prostą komunikację szeregową.

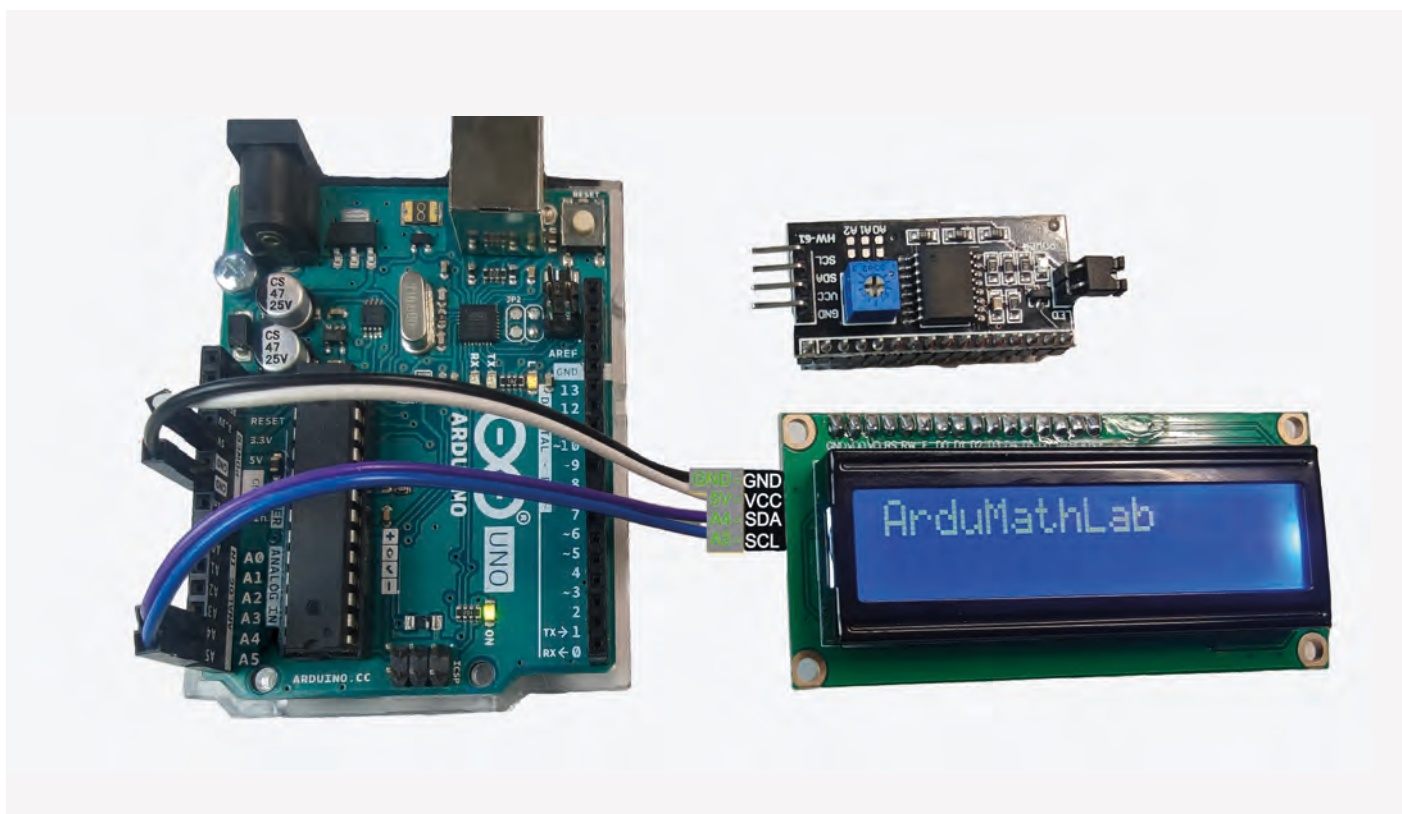
I2C (Inter-Integrated Circuit) to sposób komunikacji, który wykorzystuje tylko dwa przewody sygnałowe:

- **SDA** – do przesyłania danych,
- **SCL** – do synchronizacji (zegara).

Dzięki temu jeden mikrokontroler może obsługiwać wiele urządzeń, używając niewielkiej liczby pinów. Każde urządzenie I2C ma swój adres, który działa jak numer domu – Arduino wysyła dane dokładnie tam, gdzie trzeba.



Rysunek 33
Skomplikowane podłączenie LCD16x2 bez modułu I2C, źródło: gotowy schemat Tinkercad.



Rysunek 34
Proste podłączenie LCD16x2 z modułem I2C oraz modułem I2C znajdujący się pod wyświetlaczem LCD

Zastosowanie modułu I2C sprawia, że:

- układ jest prostszy do podłączenia,
- zmniejsza się liczba błędów,
- projekt jest bardziej czytelny i estetyczny,
- czas potrzebny na uruchomienie projektu jest krótszy.



Zadanie 1

Wyświetlanie tekstu i licznika czasu na LCD

Cel:

Nauczyć się wyświetlać tekst i zmienne na wyświetlaczu LCD 16×2 oraz zrozumieć, że program może dynamicznie aktualizować informacje widoczne na ekranie.

Polecenie:

1. Podłącz wyświetlacz LCD 16×2 z modułem I2C do Arduino zgodnie z rysunkiem.
2. W PictoBloxie utwórz program, który:
 - a. w pierwszej linii wyświetla imię Twoje i kolegi/koleżanki,
 - b. tworzy zmienną czas, która:
 - zaczyna od wartości 1,
 - zwiększa się co 1 sekundę,
 - wyświetlana jest w drugiej linii

3. Zadbaj o to, aby aktualna wartość zmiennej czas była wyświetlana na ekranie LCD i zmieniała się w czasie co 1 sekundę.

- wyczyść ekran LCD,
- ustaw pozycję kursora,
- wyświetl aktualną wartość zmiennej czas.

Kryterium osiągnięcia celu

- W pierwszej linii LCD wyświetlane są imiona.
- W drugiej linii wyświetlany jest licznik czasu zwiększający się co 1 sekundę.
- Wyświetlane wartości aktualizują się poprawnie i nie nakładają się na siebie.



1. Dlaczego konieczne jest czyszczenie ekranu i ponowne ustawianie kursora po każdej zmianie wartości zmiennej, zamiast wyświetlania samej nowej liczby?
2. Co zobaczysz na ekranie LCD, jeśli ilość znaków w wierszu przekroczy 16?

Czujnik odległości HC-SR04

Czujnik odległości (ultradźwiękowy) pozwala mierzyć, jak daleko znajduje się obiekt od czujnika. Działa na zasadzie podobnej do echolokacji, którą wykorzystują np. nietoperze. Czujnik wysyła krótki impuls ultradźwiękowy, a następnie mierzy czas, po jakim fala odbije się od przeszkody i wróci do czujnika. Na podstawie czasu powrotu sygnału Arduino oblicza odległość. Im dłużej fala wraca, tym obiekt znajduje się dalej. Czujnik nie mierzy więc bezpośrednio odległości, lecz czas, który następnie jest przeliczany na centymetry lub milimetry.



Rysunek 35
Przykłady czujników odległości

Czujnik odległości pełni rolę źródła danych liczbowych, które:

- mogą być wyświetlane na LCD,
- analizowane matematycznie,
- porównywane z progami i przedziałami.

Typowy czujnik ultradźwiękowy posiada **4 piny**:

Pin	Znaczenie	Podłączenie do Arduino
VCC	zasilanie czujnika	5V
GND	masa układu	GND
TRIG	wyzwolenie pomiaru	dowolny pin cyfrowy (np. D9)
ECHO	sygnał powrotu impulsu	dowolny pin cyfrowy (np. D10)

- Pin **TRIG** odpowiada za wystanie impulsu ultradźwiękowego.
- Pin **ECHO** zwraca impuls, którego długość zależy od odległości obiektu.
- Piny TRIG i ECHO nie są pinami analogowymi – pracują cyfrowo (stan niski / wysoki).



1. Czujnik odległości mierzy czas, a nie odległość bezpośrednio.
2. Odległość jest wynikiem obliczeń matematycznych wykonywanych przez program.





Zadanie 2

Czujnik zbliżeniowy, odczyt odległości



Rysunek XX

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

Cel:

Nauczyć się odczytywać dane z czujnika odległości HC-SR04 oraz wyświetlać wynik pomiaru w formie liczbowej i opisowej na wyświetlaczu LCD 16×2.

Polecenie:

Utwórz w PictoBlox program, który co 1 sekundę mierzy odległość od przeszkody za pomocą czujnika HC-SR04 i wyświetla wynik na ekranie LCD. Program powinien wyświetlać:

1. w pierwszej linii: aktualna odległość do przeszkody w centymetrach (np. Odlegl: 35 cm).
2. w drugiej linii: opis słowny w zależności od odległości:
 - jeśli ≤ 20 cm → blisko
 - jeśli > 20 cm i < 60 cm → średnio
 - jeśli ≥ 60 cm → daleko

Krok 1 – zasilanie.

Podłącz pin 5V z Arduino do czerwonej szyny (+) na płytce stykowej. Płytkę stykową służy tutaj wyłącznie do rozdzielenia zasilania +5V do dwóch urządzeń odbiorczych (LCD i HC-SR04).



Krok 2 – podłączenie wyświetlacza LCD 16×2 I2C

Podłącz piny wyświetlacza:

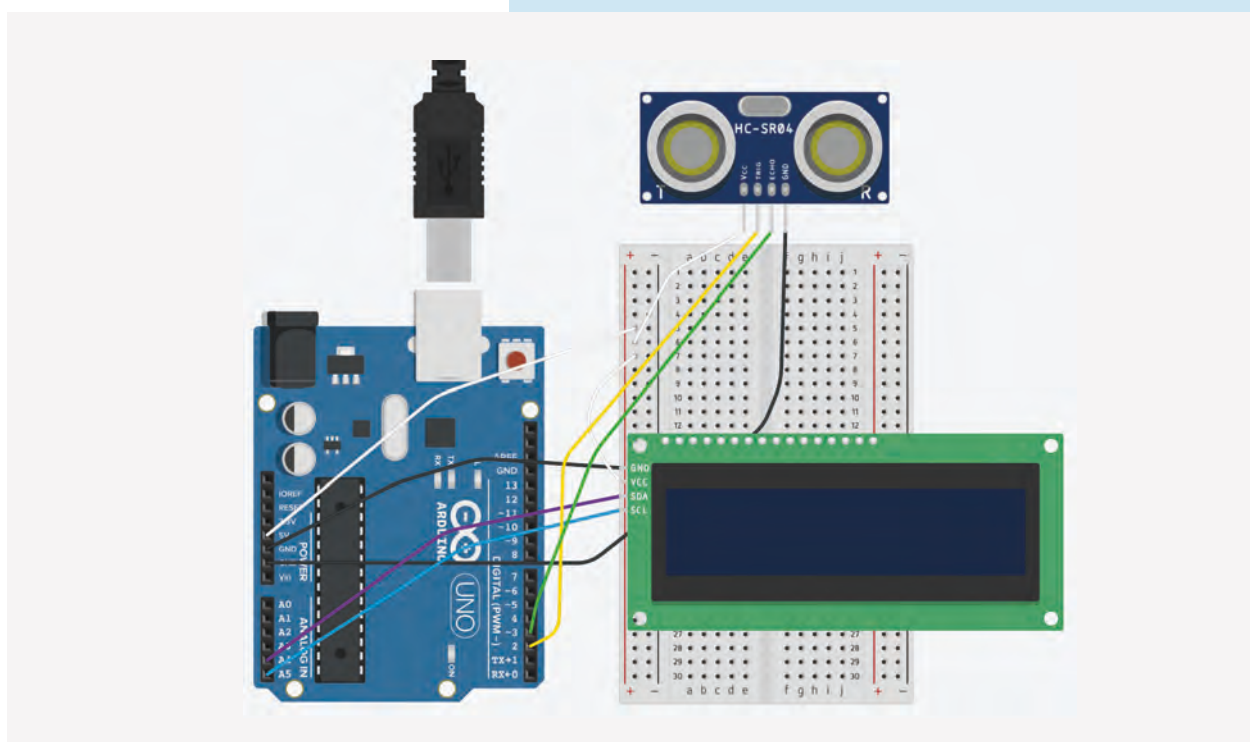
- **VCC** → szyna (+) na płytce stykowej
- **GND** → **GND Arduino**
- **SDA** → pin **A4** Arduino
- **SCL** → pin **A5** Arduino

Krok 3 – podłączenie czujnika odległości HC-SR04

Podłącz piny czujnika HC-SR04:

- **VCC** → szyna (+) na płytce stykowej
- **GND** → **GND Arduino**
- **TRIG** → pin **D2** Arduino
- **ECHO** → pin **D3** Arduino

Rysunek 36
Schemat połączenia wyświetlacza LCD16x2 I2C oraz czujnika odległości.





1. Co stanie się z wyświetlanym wynikiem, jeśli przeszkoda znajdzie się poza zasięgiem czujnika? Jak system zachowuje się w sytuacji braku wiarygodnych danych?
2. Gdy przeszkoda stoi nieruchomo, czy odczyt na LCD jest identyczny w każdej sekundzie, czy jednak lekko się waha? Jeśli tak to jak duże są różnice?

Krok 4 – Utwórz program w PictoBlox

Program powinien:

- działać w trybie Upload,
- wykonywać pomiar odległości co 1 sekundę,
- wyświetlać odległość w pierwszej linii LCD,
- wyświetlać opis „blisko”, „średnio” lub „daleko” w drugiej linii LCD,
- aktualizować dane w nieskończonej pętli.

Kryterium osiągnięcia celu

- Wyświetlacz LCD pokazuje aktualną odległość w centymetrach.
- Wyświetlany opis („blisko”, „średnio”, „daleko”) odpowiada zmierzonej odległości.
- Dane aktualizują się automatycznie co 1 sekundę.
- Wskazania zmieniają się w zależności od położenia przeszkody.

Odkrywamy matematykę: Odległość i prędkość, średnia arytmetyczna w pomiarach.

Echolokacja, ultradźwięki, Jak działa pomiar odległości w HC-SR04

Czujnik ultradźwiękowy HC-SR04 działa na zasadzie pomiaru czasu przelotu fali dźwiękowej.

Czujnik:

1. wysyła krótki impuls ultradźwiękowy,
2. fala odbija się od przeszkody,
3. czujnik mierzy czas, po jakim fala wróciła.

Ultradźwięki rozchodzą się
w powietrzu z prędkością
około: **340 m/s**



czyli:
34 cm w 0,001 s
1 cm w ok. 29 μs

Ponieważ fala:

- leci do przeszkody i wraca, zmierzony czas dotyczy drogi podwójnej.

Dlatego odległość obliczana jest według schematu:

$$\text{odległość} = (\text{czas} \times \text{prędkość dźwięku}) \div 2$$

- czujnik nie mierzy odległości,
- czujnik mierzy czas,
- odległość jest wynikiem obliczeń.





Nietoperze działają dokładnie na tej samej zasadzie:

1. emitują ultradźwięki,
2. odbierają echo,
3. na podstawie czasu powrotu oceniają:
 - odległość,
 - wielkość przeszkody,
 - a nawet jej ruch.

Różnica:

- nietoperz robi to mózgiem,
- Arduino robi to programem.



Zadanie 1

Obliczanie średniej
z pomiarów
odległości

Cel:

Obliczyć średnią arytmetyczną na podstawie rzeczywistych pomiarów wykonanych przez czujnik odległości oraz zrozumieć, jak zmiana danych pomiarowych wpływa na wartość średniej.

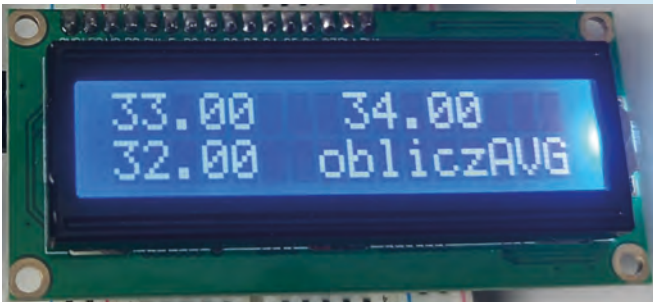
Krok 1. Przygotowanie układu

1. Upewnij się, że czujnik ultradźwiękowy i wyświetlacz LCD są podłączone tak jak w części informatycznej modułu.

2. Wgraj do Arduino program zad1.sb3 i uruchom go.

Krok 2. Odczyt danych pomiarowych

1. Po uruchomieniu programu czujnik wykona trzy kolejne pomiary odległości.
2. Wyniki pojawią się na wyświetlaczu LCD:
 - w pierwszym wierszu – dwa pierwsze pomiary,
 - w drugim wierszu – trzeci pomiar oraz komunikat **„obliczAVG”**.
3. Zapisz wszystkie trzy wartości pomiarów w zeszycie lub na karcie



Rysunek 37
wyświetlacz pokazuje 3 pomiary
odległości i pyta o średnią

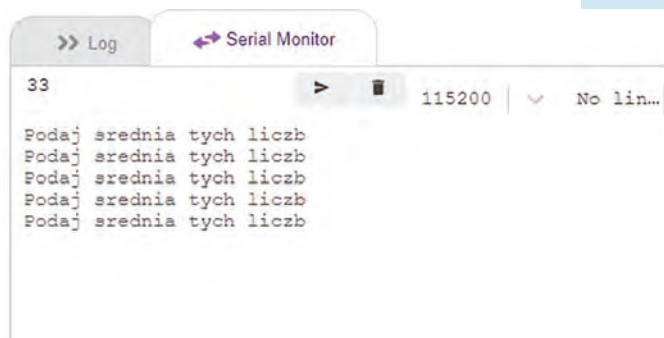
Krok 3. Obliczenie średniej arytmetycznej

1. Oblicz średnią arytmetyczną z trzech pomiarów według wzoru:

$$\text{średnia} = (\text{pomiar}_1 + \text{pomiar}_2 + \text{pomiar}_3) \div 3$$

2. Gdy program poprosi o wynik, wpisz obliczoną średnią w Monitorze portu szeregowego.
3. Zatwierdź wynik klawiszem Enter.





Rysunek 38
Okno monitora portu szeregowego,
miejsce gdzie wprowadza się wynik.

3. Krok 4. Sprawdzenie wyniku

1. Program porówna wpisaną wartość z rzeczywistą średnią.
2. Na wyświetlaczu LCD pojawi się informacja, czy wynik jest poprawny.

Kryterium osiągnięcia celu

- Program wykonuje trzy pomiary i wyświetla je na ekranie LCD.
- Uczeń zapisuje wszystkie trzy wartości pomiarów.
- Uczeń poprawnie oblicza średnią arytmetyczną z uzyskanych danych.
- Po wpisaniu wyniku program potwierdza poprawność obliczenia.

Warianty wykonania zadania 1

Wariant A (bez zmiany położenia przeszkody):

Nie zmieniaj położenia obiektu podczas pomiarów, nie ruszaj czujnikiem odległości. Uzyskane wartości powinny być bliskie. Jest to naturalne i świadczy o stabilności pomiaru. Wiele urządzeń pomiarowych zakłada pewną tolerancję błędów pomiaru.

Wariant B (ze zmianą położenia przeszkody):

Po wykonaniu pierwszych obliczeń, zrestartuj Arduino. Program zacznie ponownie obliczać odległości co 5 sekund. Po każdym pomiarze zmień położenie przeszkody (np. przesun książkę bliżej lub dalej). Zauważ, jak zmiana danych wpływa na wartość średniej.

Wariant C (Eksperymentuj ze skrajnymi wartościami):

Sprawdź, jaki wpływ na średnią arytmetyczną mają wartości skrajne. Wykonaj trzy pomiary w taki sposób, aby dwa pomiary dotyczyły przeszkody ustawionej blisko czujnika, a jeden bardzo daleko (lub odwrotnie: dwa daleko i jeden blisko). Zapisz wyniki i oblicz średnią. Zastanów się, który z pomiarów najsilniej wpływa na końcowy wynik.



1. Jeśli wszystkie trzy pomiary są bardzo podobne, czy średnia arytmetyczna różni się od pojedynczego pomiaru?
Co to mówi o stabilności pomiarów?
2. Jak zmienia się średnia, gdy jeden z pomiarów znacznie różni się od pozostałych? Czy pojedynczy skrajny wynik ma duży wpływ na końcową wartość?





Zadanie 2

Nowa średnia po
dodaniu kolejnego
pomiaru

Cel:

Wyznaczyć brakującą wartość pomiaru na podstawie znanej średniej arytmetycznej oraz sprawdzić wynik poprzez rzeczywisty pomiar wykonany czujnikiem odległości.

Krok 1. Odczyt danych z wyświetlacza

1. Uruchom program w Arduino.
2. Na wyświetlaczu LCD pojawią się dwie informacje:
 - w pierwszej linii – średnia z trzech wykonanych wcześniej pomiarów,
 - w drugiej linii – docelowa średnia z czterech pomiarów.

Krok 2. Obliczenie brakującego pomiaru

1. Na kartce oblicz, jaka musi być wartość czwartego pomiaru, aby średnia z czterech pomiarów była równa podanej wartości docelowej.
2. Wykorzystaj wzór na średnią arytmetyczną:

$$\text{średnia} = (\text{pomiar}_1 + \text{pomiar}_2 + \text{pomiar}_3 + \text{pomiar}_4) \div 4$$

3. Wyznacz brakującą wartość pomiaru₄.





Rysunek 39
Wyświetlacz pokazuje informację czy
podano poprawny wynik.

Krok 3. Sprawdzenie wyniku za pomocą czujnika

1. W tym zadaniu nie wpisujesz odpowiedzi do monitora portu szeregowego.
2. Spróbuj ustawić przeszkodę tak, aby odczyt odpowiadał obliczonej przez Ciebie wartości.
3. Zaczynj zmieniać położenie przeszkody przed czujnikiem odległości.
4. Obserwuj aktualną wartość pomiaru wyświetlaną na LCD.
5. Gdy uzyskasz właściwą wartość, program potwierdzi poprawne wykonanie zadania.

Kryterium osiągnięcia celu

- Program wyświetla średnią z trzech pomiarów oraz docelową średnią z czterech pomiarów.
- Uczeń poprawnie oblicza wartość brakującego czwartego pomiaru.
- Uczeń ustawia przeszkodę w takiej odległości, aby rzeczywisty pomiar odpowiadał obliczonej wartości.
- Program potwierdza osiągnięcie docelowej średniej.





1. Dlaczego nie wystarczy zgadywać położenia przeszkody, ale trzeba najpierw obliczyć wymaganą wartość czwartego pomiaru? Co to pokazuje o związku między średnią a wszystkimi pomiarami?
2. Co obserwujesz, gdy zbliżasz przeszkodę do czujnika o kilka centymetrów? Czy łatwo jest uzyskać dokładnie obliczoną wartość, czy wymaga to precyzyjnego ustawienia?

Czego się nauczyłeś?



Matematyka

- jak obliczać średnią arytmetyczną na podstawie wyników pomiarów,
- jak zmiana pojedynczego pomiaru wpływa na wartość średniej,
- jak obliczyć brakującą wartość pomiaru na podstawie znanej średniej,
- jak wykorzystać obliczenia matematyczne do przewidywania i weryfikowania wyników rzeczywistego pomiaru.



Informatyka

- jak wykorzystać wyświetlacz LCD 16×2 do prezentowania tekstu i danych liczbowych,
- jak odczytywać dane z czujnika ultradźwiękowego i wykorzystywać je w programie,
- jak program może cyklicznie wykonywać pomiary i aktualizować wyświetlane informacje,
- jak połączyć czujnik i wyświetlacz, aby tworzyć układ mierzący i prezentujący dane w czasie rzeczywistym.



