

Podstawy Systemów Wbudowanych

Laboratorium

Laboratorium 3: Czujniki i akwizycja danych — ADC, I²C i SPI

Autorzy: Angelika Tefelska, Dariusz Tefelski

Spis treści

1	Wprowadzenie	1
2	Wyposażenie stanowiska	1
3	Zadanie 1: Czujnik analogowy i przetwornik ADC	2
3.1	Od napięcia do liczby	2
3.2	Czujnik wilgotności gleby	2
3.3	Kalibracja i uśrednianie	4
4	Zadanie 2: Cyfrowy czujnik środowiskowy przez I2C	4
4.1	Biblioteki urządzeń	5
5	Zadanie 3: Magistrala SPI i czytnik RFID	6
5.1	Podłączenie i zasilanie czytnika RC522	7
6	Zadanie końcowe: System pomiarowy uruchamiany kartą RFID	8
7	Podsumowanie	9

1 Wprowadzenie

Cel ćwiczenia



Celem ćwiczenia jest poznanie kilku sposobów pozyskiwania danych przez system wbudowany. W pierwszej części wykorzystamy wejście analogowe i przetwornik ADC do odczytu czujnika wilgotności gleby. Następnie wrócimy do magistrali I²C poznanej w Laboratorium 2 i wykorzystamy ją do odczytu cyfrowego czujnika środowiskowego. W ostatniej części poznamy magistralę SPI na przykładzie czytnika RFID.

W poprzednim laboratorium dane docierały do mikrokontrolera poprzez UART i I²C. Tym razem porównamy trzy sytuacje: pomiar napięcia analogowego, komunikację z cyfrowym czujnikiem I²C oraz komunikację z urządzeniem SPI.

Trzy sposoby pozyskiwania informacji



	ADC	I ² C	SPI
Przykład	czujnik wilgotności gleby	ENS160/BME280	czytnik RFID
Co odbiera Arduino?	napięcie	dane cyfrowe	dane cyfrowe
Identyfikacja urządzenia	pin analogowy	adres I ² C	linia CS
Zegar transmisyjny	nie dotyczy	SCL	SCK

2 Wyposażenie stanowiska

Do wykonania ćwiczenia potrzebne są:

- Arduino UNO R4 WiFi,
- przewód USB,
- Analog Discovery II z przewodami pomiarowymi,
- analogowy czujnik wilgotności gleby Waveshare 9527,
- moduł środowiskowy SparkFun ENS160/BME280,
- przewód Qwiic do połączenia Arduino z modułem ENS160/BME280,
- czytnik RFID RC522 wraz z kartą lub brelokiem,
- płytki stykowe i przewody połączeniowe,

- dioda LED i rezystor 330 Ω ,
- przygotowana próbka suchego i wilgotnego podłoża lub inny materiał umożliwiający porównanie wskazań czujnika wilgotności.

Co należy zapamiętać?



Czytnik RFID RC522 zasilaj wyłącznie z wyjścia 3.3V Arduino. Nie podłączaj jego pinu zasilania do 5V. W stanowisku laboratoryjnym linie SPI czytnika są podłączane bezpośrednio do odpowiednich pinów Arduino zgodnie z tabelą w zadaniu 3.1.

Co należy zapamiętać?



W czujniku wilgotności gleby zanurzaj wyłącznie część pomiarową sondy. Nie zanurzaj złącza ani elementów elektronicznych. Po zakończeniu pomiarów wyjmij czujnik z wilgotnego podłoża.

3 Zadanie 1: Czujnik analogowy i przetwornik ADC

3.1 Od napięcia do liczby

Czujnik analogowy nie wysyła do Arduino gotowej liczby opisującej wielkość fizyczną. Na jego wyjściu pojawia się napięcie zależne od mierzonego zjawiska. Aby mikrokontroler mógł je przetworzyć, wykorzystuje przetwornik analogowo-cyfrowy ADC (*Analog-to-Digital Converter*).

Wartość zwracana przez ADC zależy od rozdzielczości przetwornika. Dla rozdzielczości N bitów możliwych jest 2^N poziomów cyfrowych. Przykładowo:

- 10 bitów: wartości od 0 do 1023,
- 14 bitów: wartości od 0 do 16383.

Arduino UNO R4 WiFi ma przetwornik ADC o maksymalnej rozdzielczości 14 bitów. Ze względu na zgodność z wcześniejszymi płytkami funkcja `analogRead()` domyślnie zwraca jednak wynik 10-bitowy.

Więcej bitów nie zawsze oznacza lepszy pomiar



Większa rozdzielczość oznacza mniejszy krok kwantyzacji, ale nie usuwa szumu czujnika, błędów kalibracji ani zmian napięcia zasilania. Dlatego większa liczba cyfr w wyniku nie musi oznaczać równie dużej dokładności fizycznego pomiaru.

3.2 Czujnik wilgotności gleby

Czujnik wilgotności gleby ma wyjście analogowe A0UT. Według dokumentacji producenta napięcie wyjściowe rośnie wraz ze wzrostem wilgotności badanego podłoża.

Zadanie 1.1 — Pierwszy odczyt z ADC



Podłącz czujnik do Arduino:

Czujnik wilgotności	Arduino UNO R4 WiFi
VCC	5V
GND	GND
AOUT	A0

Uruchom program:

```
1 const int SENSOR_PIN = A0;
2
3 void setup() {
4     Serial.begin(115200);
5 }
6
7 void loop() {
8     int value = analogRead(SENSOR_PIN);
9     Serial.println(value);
10    delay(200);
11 }
```

Sprawdź wskazania dla co najmniej trzech warunków, np. sondy znajdującej się w powietrzu, w suchym podłożu i w wilgotnym podłożu. Zanotuj wyniki i sprawdź, czy wraz ze wzrostem wilgotności wartość odczytu rośnie.

Następnie zweryfikuj sygnał analogowy za pomocą Analog Discovery II:

1. uruchom program *WaveForms* i otwórz przyrząd *Scope*,
2. podłącz wejście 1+ do wyjścia AOUT czujnika, a wejście 1- do GND Arduino,
3. ustaw sprzężenie DC i obserwuj poziom napięcia dla tych samych trzech warunków,
4. porównaj zmianę napięcia obserwowaną w *WaveForms* ze zmianą wartości zwracanej przez `analogRead()`.

Czy większej wartości ADC odpowiada większe napięcie na wyjściu czujnika?

Analog Discovery II jako przyrząd pomiarowy



W tym zadaniu Analog Discovery II służy wyłącznie do obserwacji sygnału z czujnika. Nie używaj wyjść generatora ani zasilaczy V+/V- do zasilania badanego układu. Arduino pozostaje zasilane z USB.

Zadanie 1.2 — Rozdzielczość 10 i 14 bitów



Dodaj w funkcji `setup()` instrukcję:

```
1 analogReadResolution(14);
```

Powtórz pomiar w tych samych warunkach. Porównaj zakres wartości dla rozdzielczości 10 i 14 bitów. Zastanów się również, czy ostatnie cyfry wyniku 14-



bitowego są całkowicie stabilne. Jeżeli nie — zaproponuj przyczynę.

3.3 Kalibracja i uśrednianie

Surowy wynik ADC nie jest jeszcze procentową wilgotnością. Czujnik wymaga kalibracji względem warunków, w których będzie używany. W prostym ćwiczeniu przyjmijmy dwie wartości odniesienia:

- `dryValue` — wynik dla przygotowanego suchego podłoża,
- `wetValue` — wynik dla przygotowanego wilgotnego podłoża.

Zadanie 1.3 — Prosta kalibracja czujnika



Wyznacz eksperymentalnie wartości `dryValue` i `wetValue`. Następnie napisz program, który przelicza surowy wynik ADC na umowną skalę od 0 do 100%:

```
1 long moisture = map(raw, dryValue, wetValue, 0, 100);  
2 moisture = constrain(moisture, 0, 100);
```

Program powinien wyświetlać jednocześnie surowy wynik ADC oraz obliczoną wilgotność podłoża w umownej skali 0–100%.

Następnie zamiast pojedynczego odczytu oblicz średnią z 10 kolejnych pomiarów. Porównaj stabilność wyniku przed i po uśrednianiu.

Co naprawdę oznacza 0–100%?



W tym ćwiczeniu nie wyznaczamy bezwzględnej zawartości wody w glebie. Skala 0–100% jest skalą względną wynikającą z dwóch punktów kalibracyjnych wybranych podczas zajęć. Inny rodzaj gleby, inne zasolenie lub inny egzemplarz sondy może dać inne wartości.

4 Zadanie 2: Cyfrowy czujnik środowiskowy przez I2C

Na ostatnich laboratoriach poznaliśmy magistralę I²C na przykładzie wyświetlacza LCD. Teraz wykorzystamy tę samą magistralę do odczytu danych z bardziej złożonego urządzenia.

Moduł SparkFun Environmental Combo Breakout zawiera dwa układy pomiarowe:

- BME280 — temperatura, wilgotność względna i ciśnienie atmosferyczne,
- ENS160 — wskaźnik jakości powietrza, TVOC oraz eCO₂.

Oba układy znajdują się na jednej płytce, ale z punktu widzenia I²C są dwoma osobnymi urządzeniami o różnych adresach.

Co należy zapamiętać?



Do połączenia przez Qwiic używamy złącza Qwiic na Arduino UNO R4 WiFi. Złącze to pracuje z napięciem 3,3 V. Nie podłączaj do niego urządzeń wymagających zasilania 5 V.

Zadanie 2.1 — Dwa urządzenia na jednej magistrali



Przy odłączonym USB połącz moduł ENS160/BME280 ze złączem Qwiic Arduino. Następnie wykorzystaj skaner I²C z Laboratorium 2.

Zanotuj wszystkie znalezione adresy. Powinieneś zobaczyć dwa urządzenia znajdujące się na jednym module. Na typowej konfiguracji są to adresy odpowiadające układom ENS160 i BME280.

4.1 Biblioteki urządzeń

Do obsługi modułu wykorzystamy biblioteki producenta dostępne w *Library Manager* Arduino IDE:

- SparkFun ENS160,
- SparkFun BME280.

Zadanie 2.2 — Praca z przykładami bibliotek



Zainstaluj w *Library Manager* biblioteki napisane przez SparkFun Electronics:

- bibliotekę ENS160 — wyszukaj hasło ENS160,
- bibliotekę BME280 — wyszukaj hasło BME280.

W dalszej części instrukcji nie znajdziesz gotowego programu. Tym razem Twoim zadaniem jest nauczyć się korzystać z przykładów dostarczonych razem z biblioteką i samemu napisać kod na ich podstawie.

W Arduino IDE wybierz *Plik* → *Przykłady* i odnajdź:

- dla BME280 przykład `Example1_BasicReadings`,
- dla ENS160 przykład `example1_basic`.

Uruchom i przeanalizuj oba przykłady. Zwróć szczególną uwagę na:

- sposób dołączania plików nagłówkowych za pomocą `#include`,
- utworzenie obiektu reprezentującego czujnik,
- uruchomienie magistrali I²C,
- inicjalizację czujnika i sprawdzenie, czy urządzenie odpowiada,
- ustawienie trybu pracy ENS160,
- funkcje służące do pobierania poszczególnych wielkości pomiarowych.

Na podstawie przeanalizowanych przykładów napisz **własny program**, który



co około 2 s wyświetla w *Serial Monitor*:

- temperaturę w °C,
- wilgotność względną w %,
- ciśnienie w hPa,
- AQI,
- TVOC w ppb,
- eCO₂ w ppm.

Program powinien również wyświetlić czytelny komunikat i zatrzymać dalsze wykonywanie, jeżeli którykolwiek z czujników nie odpowiada podczas inicjalizacji. Zwróć uwagę, że biblioteka BME280 zwraca ciśnienie w paskalach — aby otrzymać hPa, wynik należy podzielić przez 100.

Po uruchomieniu programu chuchnij na czujnik i sprawdź, czy zmieniają się temperatura lub wilgotność. Obserwuj również, jak w pierwszych minutach po włączeniu zachowują się wskazania ENS160.

Przykłady biblioteki jako dokumentacja



Po zainstalowaniu biblioteki, Arduino IDE udostępnia dostarczone przez jej autora programy demonstracyjne w menu *Plik* → *Przykłady*. Przy pierwszym użyciu nowego czujnika lub modułu warto rozpocząć właśnie od nich. Celem nie jest bezmyślne skopiowanie całego przykładu, lecz znalezienie sposobu inicjalizacji urządzenia oraz funkcji potrzebnych we własnej aplikacji.

ENS160 potrzebuje czasu



Po włączeniu ENS160 potrzebuje zwykle kilku minut na rozgrzanie. Bezpośrednio po uruchomieniu wskazania jakości powietrza mogą nie być jeszcze mierzalne. Podczas pierwszego uruchomienia nowego czujnika okres stabilizacji jest jeszcze dłuższy.

eCO₂ nie jest bezpośrednim pomiarem CO₂



ENS160 nie jest czujnikiem NDIR mierzącym stężenie CO₂ bezpośrednio. Wartość eCO₂ jest wartością równoważną szacowaną na podstawie odpowiedzi czujnika gazowego. Traktuj ją jako wskaźnik jakości powietrza, a nie laboratoryjny pomiar stężenia CO₂.

5 Zadanie 3: Magistrala SPI i czytnik RFID

SPI (*Serial Peripheral Interface*) jest synchronicznym interfejsem szeregowym. Typowe połączenie wykorzystuje cztery sygnały:

- SCK — zegar generowany przez urządzenie sterujące,

- COPI/MOSI — dane od urządzenia sterującego do peryferium,
- CIPO/MISO — dane od peryferium do urządzenia sterującego,
- CS/SS — wybór konkretnego urządzenia.

W przeciwieństwie do I²C urządzenia SPI nie są wybierane za pomocą adresu przesyłanego magistralą. Każde urządzenie ma zwykle osobną linię CS. Linie danych i zegara mogą być współdzielone.

COPI/CIPO a MOSI/MISO



W nowszej dokumentacji Arduino stosowane są nazwy COPI (*Controller Out Peripheral In*) i CIPO (*Controller In Peripheral Out*). W wielu bibliotekach oraz na modułach spotkasz jednak starsze oznaczenia MOSI i MISO. Oznaczają one te same kierunki transmisji.

Na UNO R4 WiFi wykorzystamy:

SPI	Arduino UNO R4 WiFi
SCK	D13
CIPO / MISO	D12
COPI / MOSI	D11
CS / SS	D10

5.1 Podłączenie i zasilanie czytnika RC522

W tym laboratorium wykorzystamy moduł RFID RC522 znajdujący się na stanowisku. Czytnik należy zasiląć z wyjścia 3.3V Arduino. Linie komunikacyjne SPI podłączamy bezpośrednio do odpowiednich pinów Arduino zgodnie z konfiguracją stosowaną na stanowisku laboratoryjnym.

Co należy zapamiętać?



Nie podłączaj pinu zasilania czytnika RC522 do 5V. Przed podłączeniem modułu zawsze sprawdź oznaczenia jego wyprowadzeń, ponieważ kolejność pinów może różnić się pomiędzy wersjami płytek.

Zadanie 3.1 — Podłączenie czytnika RFID przez SPI



Przy odłączonym zasilaniu połącz czytnik RC522 z Arduino zgodnie z poniższą tabelą:



RC522	Arduino UNO R4 WiFi
SDA / SS	D10
SCK	D13
MOSI	D11
MISO	D12
RST	D9
GND	GND
3.3V	3.3V
IRQ	nie podłączaj

Uwaga: w zależności od wersji modułu pin wyboru urządzenia może być opisany jako SDA, SS lub NSS.

Zadanie 3.2 — Odczyt UID karty lub breloka



Zainstaluj bibliotekę MFRC522. Otwórz przykład *readNUID* i przeanalizuj działanie programu. Następnie stwórz program, który odczyta ID breloczka RFID.

Po co jest linia CS?



Jeżeli do jednej magistrali SPI podłączonych jest kilka urządzeń, ich linie SCK, COPI i CIPO mogą być wspólne. Urządzenie sterujące wybiera aktywne peryferium, ustawiając odpowiedni sygnał CS. Dzięki temu tylko wybrane urządzenie powinno odpowiadać w danym momencie.

Zadanie 3.3 — Prosta kontrola dostępu



Wybierz UID jednego z dostępnych breloków jako identyfikator dozwolony. Zmodyfikuj program tak, aby:

- dla właściwego UID wyświetlał komunikat DOSTEP PRYZNANY,
- dla każdego innego UID wyświetlał DOSTEP ODRZUCONY,
- dla poprawnej karty włączał diodę LED na około 1 sekundę.

Porównuj kolejne bajty UID, a nie tekst wypisany w monitorze szeregowym.

6 Zadanie końcowe: System pomiarowy uruchamiany kartą RFID

W ostatnim zadaniu połączymy wszystkie trzy sposoby pozyskiwania danych używane podczas ćwiczenia: ADC, I²C i SPI.

Zadanie 4.1 — Autoryzowany odczyt stanu środowiska



Zaprojektuj program spełniający następującą specyfikację:

1. program cały czas oczekuje na kartę lub brelok RFID,
2. po odczytaniu nieautoryzowanego UID wyświetla tylko komunikat DOSTEP ODRZUCONY,
3. po odczytaniu autoryzowanego UID wykonuje jeden pomiar każdym czujnikiem i wypisuje w monitorze szeregowym:
 - względną wilgotność podłoża w procentach,
 - temperaturę i wilgotność powietrza,
 - ciśnienie atmosferyczne,
 - wartość AQI,
4. po zakończeniu raportu program ponownie oczekuje na kartę.

Podziel program na funkcje, np. `readSoilMoisture()`, `printEnvironment()` oraz `isAuthorized()`.

7 Podsumowanie

Po tych zajęciach powinieneś umieć



- wyjaśnić różnicę pomiędzy czujnikiem analogowym i cyfrowym,
- odczytać sygnał analogowy funkcją `analogRead()`,
- wyjaśnić znaczenie rozdzielczości ADC i zmienić ją funkcją `analogReadResolution()`,
- wykonać prostą kalibrację dwupunktową czujnika,
- wykorzystać Analog Discovery II do niezależnej obserwacji napięcia z czujnika analogowego,
- odnaleźć i przeanalizować przykłady dostarczone z biblioteką Arduino oraz na ich podstawie przygotować własny program,
- wykorzystać I²C do obsługi nowego czujnika,
- wskazać linie SCK, COPI/MOSI, CIPO/MISO i CS magistrali SPI,
- wyjaśnić różnicę pomiędzy adresowaniem I²C a wyborem urządzenia przez CS w SPI,
- poprawnie podłączyć i zasilić czytnik RFID RC522 oraz uruchomić komunikację SPI,
- odczytać UID nośnika RFID i wykorzystać go w prostej logice aplikacji.

Co należy zapamiętać?



Przed opuszczeniem stanowiska odłącz zasilanie, wyjmij czujnik wilgotności z podłoża, oczyść sondę jeśli jest zabrudzona oraz uporządkuj przewody i moduły.