|  |  |
| --- | --- |
| **UNIWERSYTET TECHNOLOGICZNO – HUMANISTYCZNY im. K. PUŁASKIEGO** | |
| **LABORATORIUM**: UKŁADY I SYSTEMY MIKROPROCESOROWE  **Ćwiczenie nr**: 5  **Temat ćwiczenia**: Zapoznanie się z wyświetlaczami tekstowymi LCD oraz sposobem ich podłączenia i sterowania za pomocą Arduino UNO.  **Data wykonania ćwiczenia**: 14.11.2018 **Prowadzący**: mgr. inż. Emil Sadowski  **Wydział**: Transportu i Elektrotechniki **Kierunek**: Elektrotechnika  **Rok akademicki**: 2018/2019 **Semestr**: 3 **Grupa**: 1 | |
| **Wykonawca ćwiczenia**:  Artur Grygiel, Mateusz Skuza,  Marcin Hałas, Maciej Sobczak.  Mateusz Paździór | **Ocena:** |

1. **Cel ćwiczenia**

Celem ćwiczenia było zapoznanie się z obsługą wyświetlacza LCD za pośrednictwem mikrokontrolera Arduino Uno.

1. **Przebieg ćwiczenia**

Przeprowadzone ćwiczenia laboratoryjne polegało na podłączeniu wyswietlacza lcd   
do mikrokontrolera Arduino Uno, w sposób pokazany na rys. 1, 2, 3 i wyświetleniu na nim dowolnego tekstu w każdej z 4 linii oraz ustawienie kursora w określonej pozycji. Kolejnym krokiem było wyświetlenie napisu „ temp ” wraz ze znakiem stopni celcjusza (rys. 3). Do powyższych założeń napisany został program przedstawiony na rys. 4.

W Arduino znajduje się osobna biblioteka ułatwiająca pracę z wyświetlaczami tekstowymi. Tym razem jej nazwa to LiquidCrystal. Warto w tym miejscu opisać kilka najważniejszych funkcji, a więc:

**Funkcja lcd.begin(znaki, linie)** ustawia ilość znaków i wierszy, na których będzie wyświetlany tekst. W tym przypadku wyświetlacz pozwala na 20 znaków, w każdej z 4 linii.

**Funkcja lcd.setCursor(pozycja, wiersz)** ustawia kursor w zadanej pozycji. Przykładowo zapis (0,0) oznacza początek napisu od pierwszego znaku, pierwszego wiersza. Z kolei zapis (0,1), to tekst od początku drugiej linijki.

Funkcja **lcd.backlight** to załączenie oświetlenia, a **lcd.noBacklight** to wyłączenie oświetlenia.

Najważniejsza funkcja - **lcd.print(napis)** wypisuje na wyświetlaczu tekst. Podczas wyświetlania kolejnych liter przesuwa się kursor. Dlatego kolejne wywołanie lcd.print zaczyna się od miejsca, gdzie był koniec poprzedniego tekstu. Chyba, że pomiędzy wywołaniami skorzysta się z setCursor.



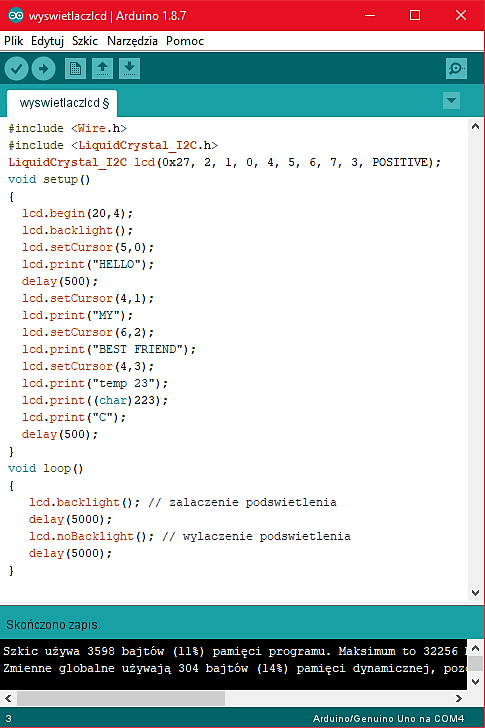
Rys. 1. Podłączenie wyświetlacza do Arduino (widok płytki z okablowaniem)



Rys. 2. Podłączenie wyświetlacza do Arduino (widok całości). Wyświetlenie komunikatu



Rys. 3. Podłączenie wyświetlacza do Arduino (widok całości). Wyświetlenie komunikatu wraz ze znakiem stopni celcjusza



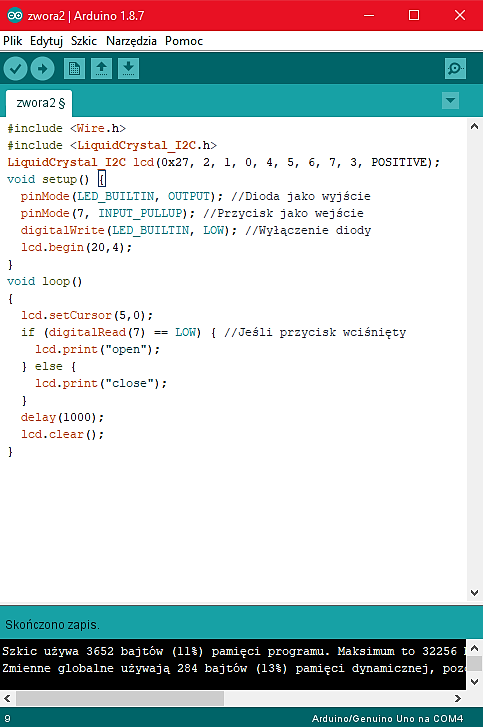
Rys. 4. Kod programu (do rys. 1, 2, 3)

W dalszej części ćwiczenia napisany został program sterujący wyświetlaniem napisu open lub close w zależności od tego, czy układ jest zwarty bądź rozwarty. W przypadku, gdy układ jest zwarty, to znaczy gdy widoczny na rysunku 4 dodatkowy pomarańczowy kabel jest jedną ze swoich dwóch końcówek umiejscowiony w pinie GND, a drugą w pinie 7 wyświetla się na ekranie wyświetlacza napis open. Gdy końcówka włożona w pin 7 zostanie wyjęta na ekranie wyświetli się napis close. Sposób podłączenia dla układu zwartego został przedstawiony na rys. 5a, natomiast dla układu rozwartego na rys. 5b. Kod programu został pokazany na rys. 6.

|  |  |
| --- | --- |
| a) | b) |
|  |  |

Rys. 5. Układ podłączenie wyświetlacza: a) zwarty, b) rozwarty

W kodzie programu na rys. 6 występuje nowa funkcja lcd.clear. Testując wyświetlanie tekstów na LCD można zauważyć, że czasami nachodzą one na siebie. Wyświetlacz nie czyści automatycznie swojej zawartości. W celu usunięcia całego tekstu należy skorzystać   
z wcześniej wspomnianej funkcji lcd.clear().



Rys. 6. Kod programu wyświetlający napis close lub open w zależności od stanu układu.

1. **Wnioski**

Na podstawie przeprowadzonego ćwiczenia laboratoryjnego można stwierdzić, że układ został poprawnie podłączony czego efektem były wyświetlane na ekranie wyświetlacza tekstowego napisy oraz znaki specjalne, to znaczy znak stopni celcjusza.

W celu przesunięcia tekstu należy ustawienić kursor w pożądanej pozycji przykładowo (5,2) oznacza to, że początek napisu jest od piątego znaku, natomiast od trzeciego wiersza. Z kolei zapis (0,1), to tekst od początku drugiej linijki. Istotne jest również, aby poprawnie zadeklarować ilość znaków i wierszy, na których będzie wyświetlany tekst. W innym przypadku tekst się nie wyświetli. Liczba znaków i wierszy jest zależna od rozmiaru samego wyświetlacza.

Podczas ćwiczenia na ekranie wyświetlacza pojawiały się niepożądane znaki, wówczas należy zastosować funkcję lcd.clear, która wyczyści ekran. Ważne jest, aby funkcja ta znajdowała się na samym końcu programu, inaczej w sposób ciągły będzie czyściła ekran.

**Literatura:**

[1]. D. Szymański, *Kurs Arduino – #7 – Wyświetlacz tekstowy, LCD 2×16* [online]. Dostępny w internecie: https://forbot.pl/blog/kurs-arduino-wyswietlacz-tekstowy-lcd-id4263.