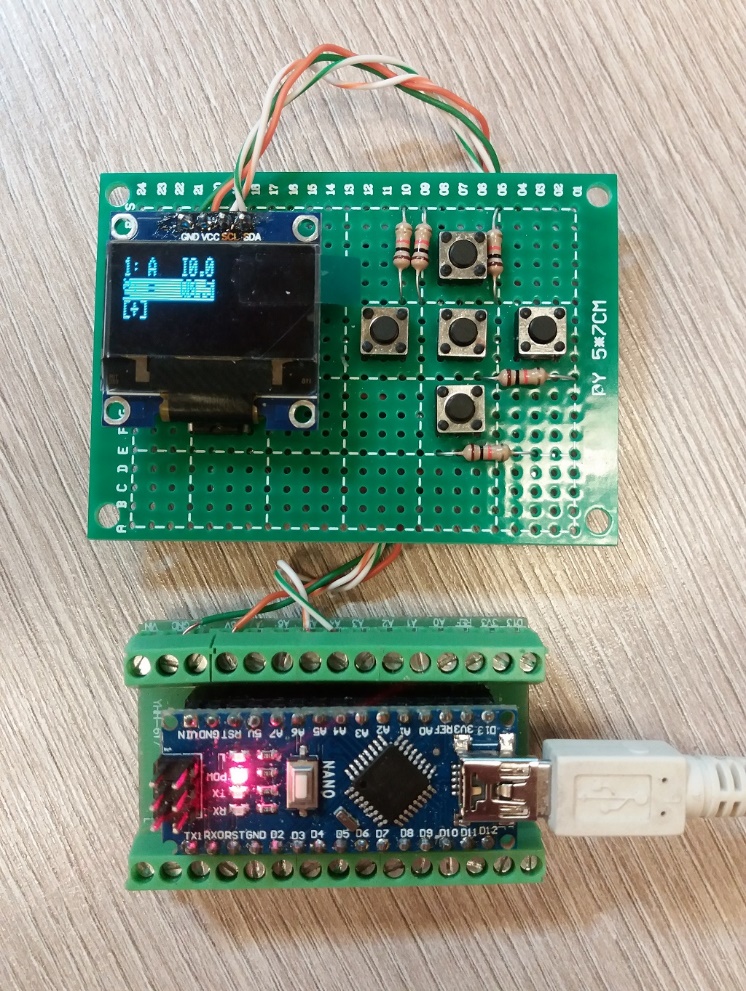
**Uważam, że mój projekt należy oznaczyć 2 gwiazdkami.**

**Arduino programowane „ręcznie”**

***Interesujące rozwiązanie umożliwiające programowanie Arduino z wykorzystaniem wyświetlacza OLED oraz klawiatury 5-przyciskowej.***



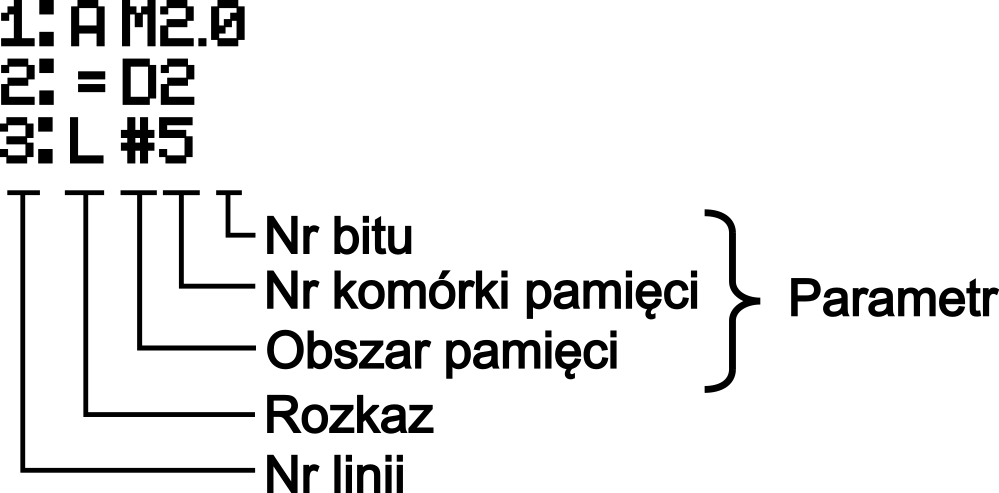
**Do czego to służy?**

Urządzenie jakim jest Arduino chyba nie trzeba nikomu przedstawiać. Za pomocą tego układu można zrealizować mnóstwo ciekawych projektów. Wystarczy kupić odpowiednią płytkę, podłączyć ją do komputera za pomocą kabla USB i można zaprogramować ją przy użyciu darmowego oprogramowania pisząc program w języku C\C++. I w tym cały problem. Otóż do zaprogramowania prostego układu potrzebujemy „potężnego komputera” i dodatkowego oprogramowania. W większości przypadków nasz układ realizuje bardzo proste zadanie i czasami chcemy wykonać drobne zmiany w jego działaniu. Czy nie da się zrobić tak, aby programowanie było realizowane już przez samo Arduino? Przedstawione rozwiązanie jest dowodem na to, że jest to możliwe. Prezentowany układ to nic innego jak płytka rozszerzająca podłączona do popularnego Arduino. Moduł wyposażony jest w wyświetlacz OLED oraz 5-przyciskową klawiaturę. Warto zwrócić uwagę na prostotę tego rozwiązania, użycie popularnych elementów oraz niski koszt.



**Jak to działa?**

Istotą działania tego urządzenia jest program wgrany jednorazowo do Arduino (program, w postaci źródłowej jest umieszczony w Elportalu wśród materiałów dodatkowych do tego numeru EdW). Program ten zajmuje się obsługą klawiatury, wyświetlacza oraz wykonywaniem zadanych przez użytkownika instrukcji. Przy pomocy 5-przyciskowej klawiatury poruszamy się po menu wyświetlanym na wyświetlaczu OLED. Wybierając odpowiednie pozycje w menu jesteśmy w stanie zaprogramować lub edytować listę instrukcji jakie ma wykonać nasze Arduino. Instrukcje te budową zbliżone są do języka asemblerowego. **Rysunek 1** przedstawia poszczególne elementy instrukcji.



Linia kodu może zawierać sam rozkaz (rozkaz bezparametrowy). Szczegółowy opis wszystkich rozkazów umieszczony jest w Elportalu wśród materiałów dodatkowych do tego numeru EdW.

**Opis budowy**

Elementem niezbędnym do prawidłowego funkcjonowania urządzenia jest wspomniany moduł rozszerzeń. Dla lepszego zrozumienia budowy i działania urządzenia, warto przeanalizować schemat ideowy, pokazany na **rysunku 2**. Układ zasadniczo składa się od z dwóch części. Pierwszą z nich jest moduł wyświetlacza OLED oznaczony na schemacie jako U2. Obecnie na rynku występuje wiele różnych tanich modułów wyświetlaczy z obsługą magistrali I2C. W tym przypadku jest to moduł wyświetlacza OLED 0.96’’ 128x64px oparty na sterowniku SSD1306. Drugim elementem modułu jest klawiatura składająca się z układu PCF8574P (8-bitowego ekspandera wejść cyfrowych z obsługą komunikacji I2C), przycisków S1…S5 i rezystorów R1…R5. Dodatkowo na module umieszczono LED D1 podłączony do wyjścia cyfrowego D13 za pośrednictwem rezystora R9.

**Lista podzespołów:**

R1…R8 10k Ω

R9 680Ω

D1 LED zielona 3mm

S1…S5 Przycisk

U1 PCF8574P

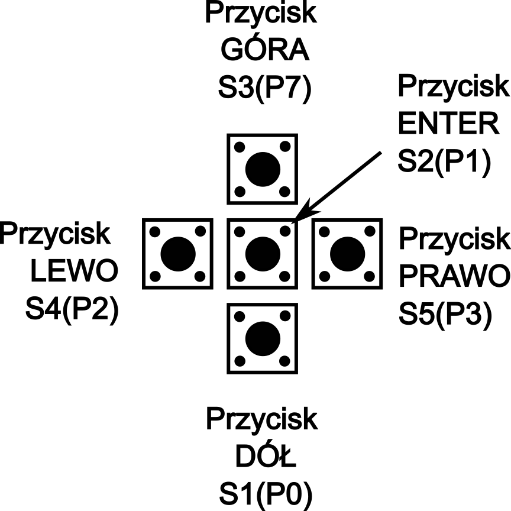
U2 OLED 0,96 oparty na SSD1306

Klawiatura i wyświetlacz podłączone są do wspólnej magistrali I2C wyprowadzonej z modułu wyjściami SCL i SDA – na płytce Arduino oznaczone jako A4, A5). Cały moduł zasilany jest z Arduino. Wejścia A0, A1, A2 układu U1 podłączone są do GND w celu ustawienia domyślnego adresu magistrali I2C układu PCF8574P. Rezystory R1, R2 pełnią funkcję podciągającą linie SDA i SCL do stanu wysokiego.



Rysunek 3. Schemat edycji przykładowego programu

**Montaż i uruchomienie**

****Wszystkie elementy modułu należy podłączyć zgodnie ze schematem ideowym, pokazanym na rysunku 2**. Uwaga: Należy zwrócić uwagę na odpowiednie podłączenie sygnałów przycisków klawiatury.** Zapewnia to prawidłowe funkcjonowanie układu.W przeciwnym wypadku niektóre przyciski nie będą działały lub będą działały nieprawidłowo. **Rysunek 3** przedstawia rozmieszenie przycisków, ich funkcję oraz odpowiednie wejście do układu PCF8574P.

Kolejną rzeczą, na którą należy zwrócić uwagę jest wyprowadzenie pinów z modułu OLED. Musi być zgodne z poniższym zdjęciem nr (kolejność pinów od lewej strony patrząc z góry: GND, VCC, SCL, SDA). Na rynku występują moduły z inną kolejnością wyprowadzeń.

Po podłączeniu modułu do płytki Arduino (w tym przypadku Arduino UNO), wgraniu programu do mikrokontrolera za pomocą oprogramowania Arduino i podłączeniu zasilania na wyświetlaczu OLED powinien pojawić się napis „NO PROGRAM”. Po 3 sekundach napis znika i pojawia się menu główne z opcjami RUN, EDIT, PROGRAM, CLEAR. Po menu poruszamy się przyciskami GÓRA, DÓŁ, a wybraną opcję aktywujemy przyciskiem ENTER (aby wrócić poziom wyżej wciskamy przycisk LEWO). Schemat na **rysunku 4** przedstawia lista kroków jakie należy wykonać, aby stworzyć prosty program mrugający diodą LED na płytce Arduino (wyjście D13 – LED\_BUILDIN) z częstotliwością 1Hz w momencie, gdy przyciśniemy przycisk DÓŁ na klawiaturze.

Każdy bloczek na rysunku przedstawia widok z wyświetlacza w danym momencie edycji. Napis zakreślony na czarno to opcja wybrana przyciskami GÓRA/DÓŁ. Strzałki wskazują na kolejny ekran jaki pojawia się po wybraniu opcji przyciskając klawisz ENTER (z wyjątkiem opisanych strzałek w pierwszej i ostatniej linii). Na początku (patrząc od góry) widać jak wyświetla się wspomniany wcześniej komunikat „No program” i po 3 sekundach przeniesieni zostajemy do menu głównego. Wybieramy EDIT i możemy dodawać kolejne linie programu „[+]”. Wybieramy wskazane na rysunku pozycji i zatwierdzamy przyciskiem „ENTER”. Na ekranach z napisami „Enter byte nr:”, „Enter bit nr:” edytujemy wartości parametrów poprzez wybranie pozycji danej cyfry przyciskami LEWO/PRAWO, a następnie zwiększanie lub zmniejszanie jej wartości przyciskami GÓRA/DÓŁ i zatwierdzamy przyciskiem „ENTER”. Strzałka zawracająca na rysunku oznacza zakończenie edycji nowej linii i powrót do listingu całego programu. Na dole rysunku z lewej strony widać cały listing programu. Aby teraz uruchomić stworzony program wychodzimy z edycji przyciskiem LEWO i przechodzimy do menu głownego. Wybieramy „Run”, a następnie „Save&Run”. Program zostaje zapisany do nieulotnej pamięci wewnętrznej i rozpoczyna się jego wykonanie. W poniższej tabeli znajduje się opis powyższego programu.

**Uwagi końcowe**

Arduino zapamiętuje program po wyłączeniu zasilania. Przy podaniu zasilania jeśli zostanie wykryty program automatycznie jest wykonywany. Aby wyjść z trybu „Running…” do menu głównego należy jednocześnie wcisnąć przyciski LEWO, PRAWO i trzymać przez 3 sekundy. W trybie „Running…” wyświetlacz wyłącza się automatycznie po 30 sekundach.

Szczegółowy opis całego projektu, lista dostępnych instrukcji, schemat poruszania się po menu urządzenia znajdują się w materiałów dodatkowych do tego numeru EdW. Większość przykładów w celu prezentacji rezultatów wykorzystuje wbudowaną w Arduino LED (na module jest to LED podłączony do wyjścia D13) oraz przyciski i wyświetlacz niniejszego modułu. Nie trzeba więc podłączać dodatkowych urządzeń, aby rozpocząć zabawę z urządzeniem.

Tabela 1. Opis programu

|  |  |
| --- | --- |
| **Linia programu** | **Opis** |
| **1: A M0.3** | Załaduj stan bitu 3 w bajcie nr 0 (specjalny bajt pamięci, którego bity zmieniają swój stan po ustalonym czasie. W tym przypadku bit 3 zmienia stan co 500ms) |
| **2: A M2.0** | Iloczyn logiczny załadowanego wcześniej stanu i bitu 0 w bajcie nr 2 (specjalny bajt zawierający stan wszystkich przycisków, w tym przypadku bit 0 odpowiada za przycisk DÓŁ) |
| **3: = D13** | Przepisz wynik powyższego iloczynu logicznego do wyjścia cyfrowego nr 13 (wbudowany LED na płytce Arduino). |



Miłosz Gołębiewski

milgo@o2.pl