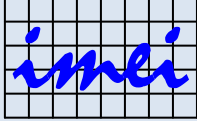


TECHNIKA MIKROPROCESOROWA Laboratorium Elektrotechnika, studia stacjonarne pierwszego stopnia	 Instytut Metrologii, Elektroniki i Informatyki
Temat: Obsługa znakowego wyświetlacza LCD	
Opracowanie instrukcji: dr inż. Mirosław Kozioł	Ćwiczenie 5

Cel ćwiczenia

Celem ćwiczenia jest zapoznanie studentów z obsługą znakowego wyświetlacza LCD pracującego pod kontrolą sterownika HD44780 firmy Hitachi dołączonego bezpośrednio do linii portów mikrokontrolera rodziny MCS-51. Ćwiczenie ma również na celu rozwinięcie umiejętności programowania strukturalnego, gdyż jego efektem powinna być biblioteka funkcji ułatwiających obsługę wspomnianego wyświetlacza.

Przygotowanie do ćwiczenia


Przed przystąpieniem do ćwiczenia należy przygotować odpowiedzi na poniższe pytania.

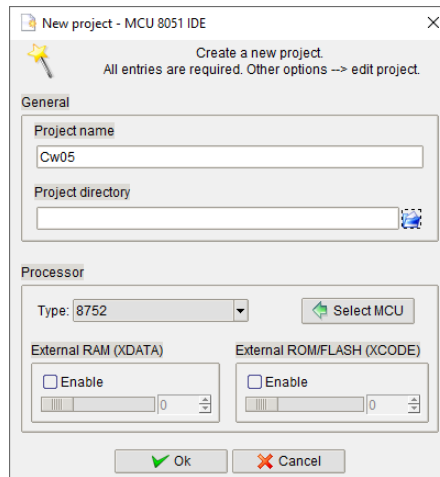
1. Jaką informację dla sterownika wyświetlacza LCD niesie stan linii RS i R/W?
2. Które z instrukcji programujących wyświetlacz LCD są wymagane do jego poprawnej inicjalizacji?
3. W jaki sposób dokonać sprawdzenia zajętości sterownika wyświetlacza LCD przed wysłaniem do niego kolejnej informacji?
4. Jak w języku C oznaczany jest koniec ciągu znakowego?
5. Jakie działania matematyczne należałoby wykonać, aby wydzielić z trzycyfrowej liczby w zapisie dziesiętnym cyfrę setek, dziesiątek i jedności?

Program ćwiczenia

Wykorzystując środowisko MCU 8051 IDE napisz program w języku C implementując kolejno przedstawione poniżej zadania. Podczas pisania programów zastosuj zasady programowania strukturalnego.

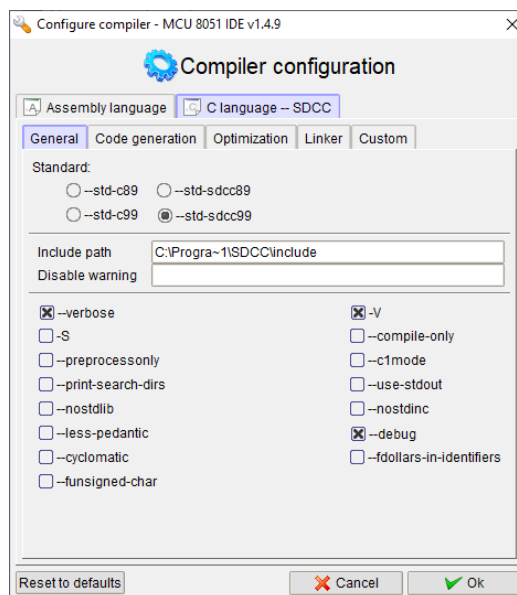
Zadanie 1

- a) W środowisku MCU 8051 IDE stwórz nowy projekt wprowadzając w oknie *New Project* ustawienia, jak na rysunku 1. W polu *Project directory* należy wpisać ścieżkę do folderu, w którym będą zapisywane pliki projektu. Można ją również wskazać z wykorzystaniem okna wyboru przez kliknięcie klawisza z ikoną .



Rys.1. Widok okna *New project* z zalecanymi ustawieniami w części *Processor*

- b) Z menu *Configure* wybierze *Configure Compiler*. W otwartym oknie, na zakładce *C language – SDCC* wybierze opcję `--std-sdcc99` oraz w polu *Include path* wpisze ścieżkę dostępu do folderu `include` znajdującego się w folderze instalacji kompilatora SDCC. Jeśli ścieżka do folderu instalacji kompilatora SDCC zawiera nazwy dłuższe niż 8 znaków, to może być konieczna jej specyfikacja jak na rysunku 2.



Rys.2. Ustawienia na zakładce *C language – SDCC*

- c) W oknie edycji kodu wpisze szablon kodu programu podany na listingu 1. Komentarze wskazują miejsca, w których powinny znaleźć się poszczególne elementy programu.

Listing 1. Szablon kodu programu

```
//=====
// Pliki nagłówkowe
#include <8052.h>

//=====
// Deklaracje zmiennych globalnych

//=====
```

```

// Prototypy funkcji

//=====
// Funkcje obsługi przerwań

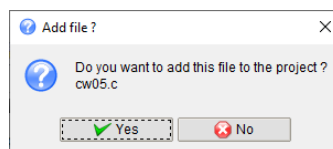
//=====
// Główna funkcja programu
void main (void)
{

    while (1);
}

//=====
// Funkcje użytkownika

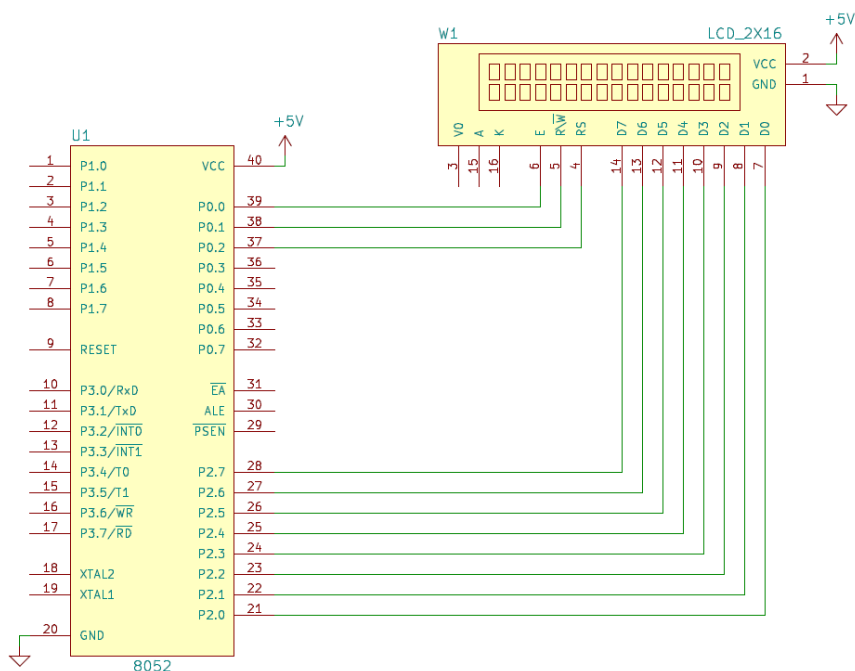
```

d) Zapisz plik z kodem. W tym celu z menu *File* wybierz *Save as*. Wprowadzając nazwę pliku staraj się nie używać polskich znaków narodowych i spacji oraz ograniczyć się do 8 znaków. Plik zapisz z rozszerzeniem *.c. Po zatwierdzeniu zapisu pliku pojawi się oknie, jak na rysunku 3, w którym należy kliknąć klawisz *Yes*.



Rys.3. Widok okna z pytaniem, czy dołączyć zapisany plik do projektu

e) Na podstawie schematu ideowego przedstawionego na rysunku 4, dokonaj podłączenia wirtualnego wyświetlacza LCD z polem odczytowym o 2 wiersza i 16 kolumnach do wyprowadzeń mikrokontrolera 8052. W tym celu wykorzystaj informacje podane w końcowej części tej instrukcji.



Rys.4. Schemat podłączenia wyświetlacza LCD do mikrokontrolera

Zadanie 2

- a) Wykorzystując informacje z wykładu do niniejszego przedmiotu napisz funkcję, która umożliwi przesyłanie do sterownika wyświetlacza LCD bajta informacji. Funkcja taka powinna mieć dwa parametry wejściowe. Pierwszy z nich powinien podawać do funkcji wartość bajta do przesłania, a drugi określać, czy informacja ma być zapisywana do rejestrów sterujących ($RS = 0$) czy też do rejestrów danych ($RS = 1$) sterownika wyświetlacza.
- b) Wykorzystując informacje z wykładu do niniejszego przedmiotu napisz funkcję, która umożliwi odczyt ze sterownika wyświetlacza LCD bajta informacji. Funkcja taka powinna mieć jeden parametr wejściowy, który powinien określać, czy odczyt będzie dotyczył rejestrów sterujących ($RS = 0$) czy też do rejestrów danych ($RS = 1$) sterownika wyświetlacza. Odczytany ze sterownika bajt informacji powinien być zwracany przez tą funkcję. Przy tworzeniu tej funkcji weź pod uwagę informacje przedstawione w końcowej części tej instrukcji oraz fakt, że linie portów w mikrokontrolerach rodziny MCS-51 pracują jako wejściowe, jeśli wcześniej zapisano do nich wartość 1.
- c) Wykorzystując funkcje zapisu i odczytu do/z sterownika wyświetlacza LCD napisz funkcję, która skonfiguruje sterownik wyświetlacza do pracy. W celu wysłania właściwych rozkazów weź pod uwagę informacje przedstawione na wykładzie z tego przedmiotu oraz na końcu niniejszej instrukcji. Rozkaz *Display on/off control* należy wysłać z takimi ustawieniami, aby włączyć kursor, a znak kursora był migający. Funkcję odczytu informacji ze sterownika wyświetlacza należy wykorzystać do sprawdzania bitu zajętości przed każdorazowym wysłaniem informacji do sterownika.
- d) Stworzoną w punkcie c) funkcję wywołaj w funkcji `main`. Następnie za tą funkcją umieść wywołanie funkcji wysyłania informacji do sterownika z takimi parametrami, które spowodują przesłanie kodu dowolnego znaku do wyświetlenia. Dokonaj kompilacji programu przez kliknięcie klawisza z ikoną . W przypadku wystąpienia błędów składniowych usuń je i przeprowadź ponowną kompilację.
- e) Sprawdź poprawność działania programu uruchamiając go w symulatorze wbudowanym w środowisko MCU 8051 IDE. W tym celu posłuż się informacjami przedstawionymi w końcowej części instrukcji. W przypadku nieprawidłowego działania programu zlokalizuj i usuń błędy, które takie działanie powodują.
- f) Po stwierdzeniu, że program działa prawidłowo, zrób zrzut ekranu okna wyświetlacza LCD w celu późniejszego wstawienia go do sprawozdania z ćwiczenia.

Zadanie 3


- a) Rozwiń tworzony program o funkcję umożliwiającą określania wiersza i kolumny, od których rozpocznie się wyświetlanie znaków na wyświetlaczu LCD. W tym celu tworzona funkcja powinna odpowiednio uzupełniać siedem najniższych bitów rozkazu *Set DDRAM address* adresem w pamięci DDRAM odpowiadającym polu znaku występującemu w danym wierszu i kolumnie, a następnie wysłać go do sterownika wyświetlacza. Informacje o numerze kolumny i wiersza powinny być podawane, jako parametry wejściowe tworzonej funkcji. Przy tworzeniu tej funkcji należy przyjąć, że kolumny i wiersze są numerowane, jak na rysunku 5.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F


Przyjęta numeracja kolumn

Przyjęta numeracja wierszy

Rys.5. Przyjęta numeracja wierszy i kolumn w wyświetlaczu LCD o 2 wierszach i 16 kolumnach. W kratkach podano adres w pamięci DDRAM odpowiadające poszczególnym polom znakowym

- Stworzoną funkcję wywołaj w funkcji `main` po funkcji inicjalizacji sterownika, a przed funkcją wyświetlającą znak. Parametry wywołania tej funkcji przyjmij inne niż 1,1 (kolumna 1 wiersz 1).
- Dokonaj kompilacji programu przez kliknięcie klawisza z ikoną . W przypadku wystąpienia błędów składniowych usuń je i przeprowadź ponowną kompilację.
- Sprawdź poprawność działania programu uruchamiając go w symulatorze wbudowanym w środowisko MCU 8051 IDE. W przypadku nieprawidłowego działania programu zlokalizuj i usuń błędy, które takie działanie powodują.
- Po stwierdzeniu, że program działa prawidłowo, zrób zrzut ekranu okna wyświetlacza LCD w celu późniejszego wstawienia go do sprawozdania z ćwiczenia.


Zadanie 4

- Rozwiń tworzony program o funkcję umożliwiającą prezentację na wyświetlaczu LCD ciągu znakowego, do którego wskaźnik będzie parametrem wejściowym tej funkcji. Podczas tworzenia kodu tej funkcji weź pod uwagę, jak w języku C jest oznaczany koniec ciągu znakowego.
- Stworzoną funkcję wywołaj w funkcji `main` po funkcji przejścia do określonego wiersza i kolumny, zastępując nią funkcję wyświetlającą pojedynczy znak. Jako parametr wywołania tej funkcji przyjmij dowolny ciąg znakowy, np. swoje imię.
- Dokonaj kompilacji programu przez kliknięcie klawisza z ikoną . W przypadku wystąpienia błędów składniowych usuń je i przeprowadź ponowną kompilację.
- Sprawdź poprawność działania programu uruchamiając go w symulatorze wbudowanym w środowisko MCU 8051 IDE. W przypadku nieprawidłowego działania programu zlokalizuj i usuń błędy, które takie działanie powodują.
- Po stwierdzeniu, że program działa prawidłowo, zrób zrzut ekranu okna wyświetlacza LCD w celu późniejszego wstawienia go do sprawozdania z ćwiczenia.


Zadanie 5

- Rozwiń tworzony program o funkcję umożliwiającą prezentację na wyświetlaczu LCD liczby w zapisie dziesiętnym, która składa się maksymalnie z trzech cyfr. Wartość tej liczby powinna być wprowadzana do funkcji jako parametr wejściowy typu `unsigned char`. Pamiętaj, że po uzyskaniu cyfr liczby a przed ich wysłaniem do sterownika wyświetlacza należy zamienić je na kody odpowiednich znaków prezentujących te cyfry.
- Stworzoną funkcję wywołaj w funkcji `main` po funkcji wyświetlającej ciąg znaków (ciąg znaków powinien być wyświetlany w górnej linii wyświetlacza LCD, a liczba w dolnej).

Pamiętaj, aby parametr, z jakim wywołasz funkcję wyświetlającą liczbę nie przekroczył zakresu liczb typu `unsigned char`.

- c) Dokonaj kompilacji programu przez kliknięcie klawisza z ikoną . W przypadku wystąpienia błędów składniowych usuń je i przeprowadź ponowną kompilację.
- d) Sprawdź poprawność działania programu uruchamiając go w symulatorze wbudowanym w środowisko MCU 8051 IDE. W przypadku nieprawidłowego działania programu zlokalizuj i usuń błędy, które takie działanie powodują.
- e) Po stwierdzeniu, że program działa prawidłowo, zrób zrzut ekranu okna wyświetlacza LCD w celu późniejszego wstawienia go do sprawozdania z ćwiczenia.

Zadanie 6

- a) Do stworzonego programu dopisz funkcję, która realizuje programowanie własnego znaku o wybranym przez siebie kształcie. Kształt ten nie może być abstrakcyjny, tylko powinien coś symbolizować. Sprawozdania oddane z tym samym znakiem będą traktowane jako plagiat.
- b) Stworzoną funkcję wywołaj w funkcji `main` po funkcji wyświetlającej liczbę. Własny znak zaprezentuj w tej samej linii co liczbę, ale na prawo od niej w odstępie jednego pustego pola znaku.
- c) Dokonaj kompilacji programu przez kliknięcie klawisza z ikoną . W przypadku wystąpienia błędów składniowych usuń je i przeprowadź ponowną kompilację.
- d) Sprawdź poprawność działania programu uruchamiając go w symulatorze wbudowanym w środowisko MCU 8051 IDE. W przypadku nieprawidłowego działania programu zlokalizuj i usuń błędy, które takie działanie powodują.
- e) Po stwierdzeniu, że program działa prawidłowo, zrób zrzut ekranu okna wyświetlacza LCD w celu późniejszego wstawienia go do sprawozdania z ćwiczenia.

Sprawozdanie z ćwiczenia

Sprawozdanie z ćwiczenia powinno być dostarczone prowadzącemu zajęcia w określonej przez niego formie (pisemnej lub elektronicznej) i zawierać następujące elementy.

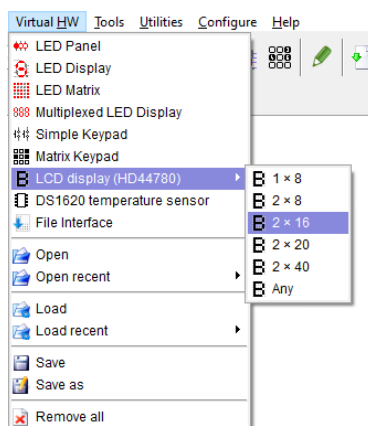
- W przypadku zadania 2 należy zamieścić kod funkcji odpowiedzialnych za odczyt i zapis informacji z/do sterownika wyświetlacza LCD oraz funkcji przeprowadzającej jego inicjalizację. Kod tych funkcji powinien być opatrzony komentarzami i dla każdej funkcji powinien być zamieszczony algorytm jej działania. Należy również zamieścić wykonany zrzut ekranu okna wyświetlacza LCD po realizacji kodu programu. W postaci opisu tekstowego należy przedstawić informacje, jakie rozkazy zostały użyte przy inicjalizacji sterownika wyświetlacza i jak ustawiono ich parametry.
- W przypadku zadania 3 należy zamieścić kod funkcji przewidzianej do napisania w tym zadaniu opatrzony komentarzami oraz algorytm jej działania. Należy również zamieścić wykonany zrzut ekranu okna wyświetlacza LCD po realizacji kodu programu oraz przez opis tekstowy wyjaśnić, w jaki sposób dokonano zamiany numeru wiersza i kolumny na rozkaz wysłany do sterownika wyświetlacza.
- W przypadku zadania 4 należy zamieścić kod funkcji przewidzianej do napisania w tym zadaniu opatrzony komentarzami oraz algorytm jej działania. Należy również zamieścić wykonany zrzut ekranu okna wyświetlacza LCD po realizacji kodu programu oraz wyjaśnić, w jaki sposób dokonano detekcji końca ciągu znakowego do wyświetlenia.

- W przypadku zadania 5 należy zamieścić kod funkcji przewidzianej do napisania w tym zadaniu opatrzone komentarzami oraz algorytm jej działania. Należy również zamieścić wykonany zrzut ekranu okna wyświetlacza LCD po realizacji kodu programu oraz przez opis tekstowy wyjaśnić idee, według której zrealizowano wydzielenie cyfry setek, dziesiątek i jedności oraz zamianę cyfr na kody wyświetlacza LCD.
- W przypadku zadania 6 należy zamieścić kod funkcji main i funkcji przeprowadzającej definiowanie własnego znaku (obydwie opatrzone komentarzami) oraz algorytm ich działania. Należy również zamieścić wykonany zrzut ekranu okna wyświetlacza LCD po realizacji kodu programu oraz przez opis tekstowy wyjaśnić co prezentuje zdefiniowany znak, oraz jak wykonano jego definiowanie.

Informacje dotyczące komentowania kodu programu znajdują się w dalszej części instrukcji, natomiast informacje dotyczące tworzenia schematów blokowych algorytmów pracy programów znajdują się w instrukcji do ćwiczenia 3.

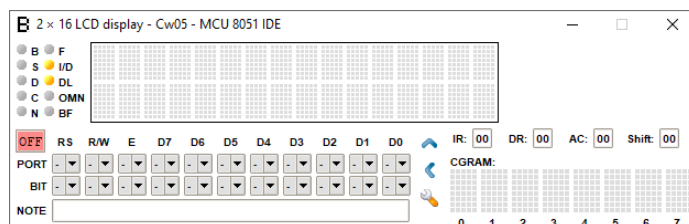
Wirtualny model wyświetlacza LCD ze sterownikiem HD44780 w środowisku MCU 8051 IDE

W środowisku MCU 8051 IDE istnieje możliwość obserwacji działania wyświetlacza znakowego LCD ze sterownikiem HD44780 podczas symulacji wykonywania kodu programu realizującego jego obsługę, gdy wyświetlacz ten jest podłączony bezpośrednio do linii portu. Aby móc wizualnie obserwować reakcję takiego wyświetlacza, należy z menu *Virtual HW* wybrać pozycję zaznaczoną na rysunku 6.



Rys.6. Wybór okna prezentującego zachowanie wyświetlacza LCD o 2 wierszach i 16 kolumnach

Spowoduje to uwidocznienie okna przedstawionego na rysunku 7.



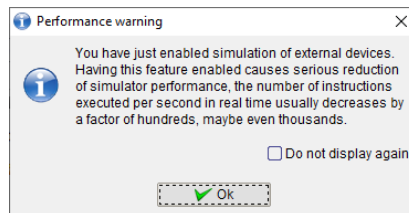
Rys.7. Widok okna wyświetlaczu LCD o 2 wierszach i 16 kolumnach

W głównej części tego okna znajduje się pole otoczone czarną ramką, które odpowiada polu odczytowemu wyświetlacza LCD o 2 wierszach i 16 kolumnach. W lewej części okna uwidoczniiony

jest stan bitów ustawianych przez wysyłane do sterownika wyświetlacza rozkazy. Kolor żółty oznacza, że dany bit jest ustawiony w stan wysoki, a szary, że w stan niski.


W lewej dolnej części okna realizujemy przyporządkowanie linii sterownika tego wyświetlacza do odpowiednich linii portu mikrokontrolera. W prawej dolnej części okna znajduje się 8 pól, które obrazują kształt znaków zdefiniowanych przez użytkownika.

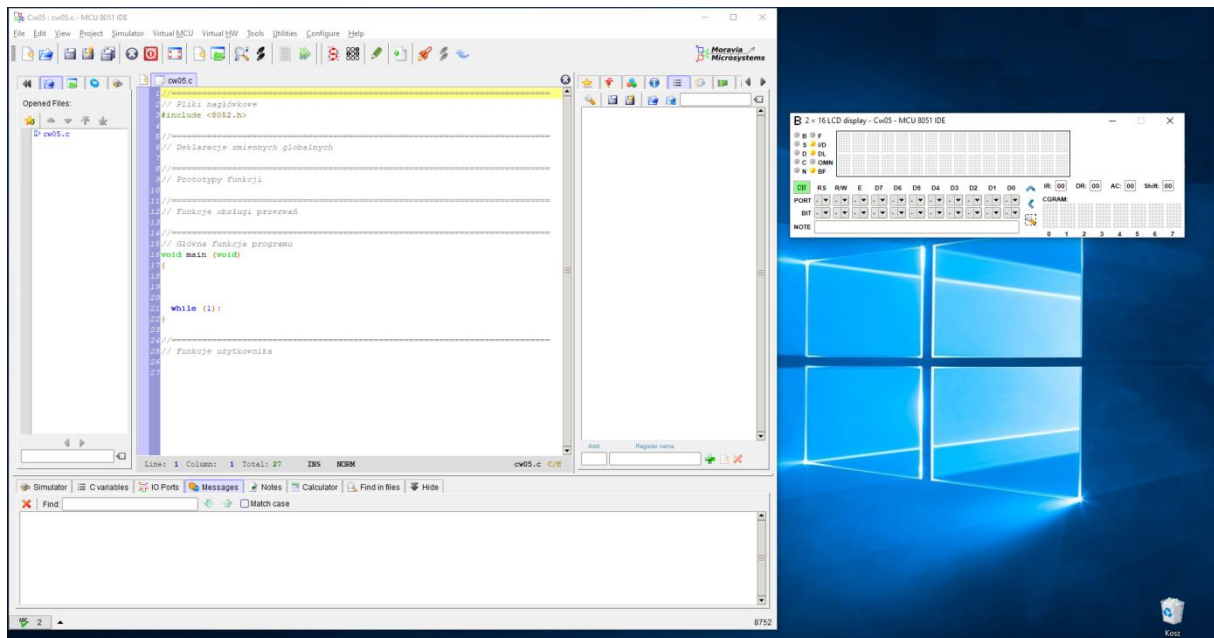
Domyślnie wyświetlacz jest wyłączony. Aby go włączyć, należy kliknąć klawisz **OFF**. Spowoduje to pojawienie się okna informacyjnego, jak na rysunku 8. W oknie tym należy kliknąć klawisz **Ok**.




Rys.8. Okno informacyjne po pierwszym włączeniu symulacji wyświetlacza LCD

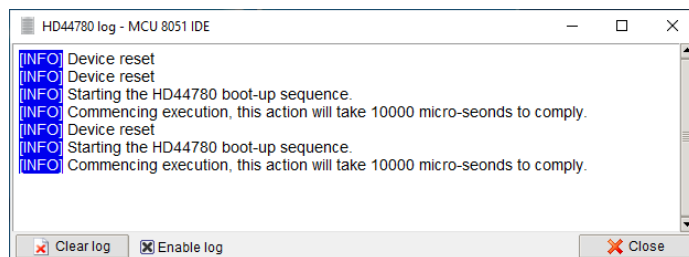
Po tych działaniach klawisz **OFF** przyjmie postać **ON**.

Podczas realizacji tych operacji można zauważyć, że okno wyświetlacza LCD cały czas chowa się „pod spód”. Aby temu zapobiec, należy w oknie symulacji wyświetlacza LCD kliknąć klawisz z ikoną  i z rozwiniętego menu wybrać ostatnią pozycję, tj. *Window always on top*. Można również zmniejszyć główne okno programu MCU 8051 IDE tak, aby zajmowało tylko połowę pulpitu Windows, a okno wyświetlacza LCD umieścić na drugiej połowie tego pulpitu, jak na rysunku 9.



Rys.9. Przykładowe ustawienie okna programu MCU 8051 IDE oraz okna wyświetlacza LCD

Warto również uaktywnić okno, w którym prezentowane są m.in. informacje, czy dana operacja odczytu lub zapisu z/do wyświetlacza została przeprowadzona poprawnie. W tym celu w oknie symulacji wyświetlacza LCD należy kliknąć klawisz z ikoną  i z rozwiniętego menu wybrać *Show HD44780 log*. Spowoduje to uwidocznienie okna jak na rysunku 10.



Rys.10. Okno z informacjami dotyczącymi wyświetlacza LCD

Należy mieć na uwadze, że w pewnych aspektach program realizujący obsługę znakowego wyświetlacza LCD dostępnego w środowisku MCU 8051 IDE wymaga odpowiedniej realizacji.

Zgodnie z dokumentacją sterownika HD4470, której fragment przedstawiono na rysunku 11, po włączeniu zasilania realizuje on procedurę inicjalizacyjną, która m.in. polega na wykonaniu czterech rozkazów z pewnymi domyślnymi ustawieniami. Jak widać z przedstawionego na rysunku 11 fragmentu dokumentu, cała ta procedura trwa ok. 10 ms. W tym czasie nie powinny być do niego realizowane żadne odniesienia.


Initializing by Internal Reset Circuit

An internal reset circuit automatically initializes the HD44780U when the power is turned on. The following instructions are executed during the initialization. The busy flag (BF) is kept in the busy state until the initialization ends (BF = 1). The busy state lasts for 10 ms after V_{CC} rises to 4.5 V.

1. Display clear
2. Function set:
 - DL = 1; 8-bit interface data
 - N = 0; 1-line display
 - F = 0; 5 × 8 dot character font
3. Display on/off control:
 - D = 0; Display off
 - C = 0; Cursor off
 - B = 0; Blinking off
4. Entry mode set:
 - I/D = 1; Increment by 1
 - S = 0; No shift

Note: If the electrical characteristics conditions listed under the table Power Supply Conditions Using Internal Reset Circuit are not met, the internal reset circuit will not operate normally and will fail to initialize the HD44780U. For such a case, initialization must be performed by the MPU as explained in the section, Initializing by Instruction.

Rys.11. Fragment dokumentacji sterownika HD44780

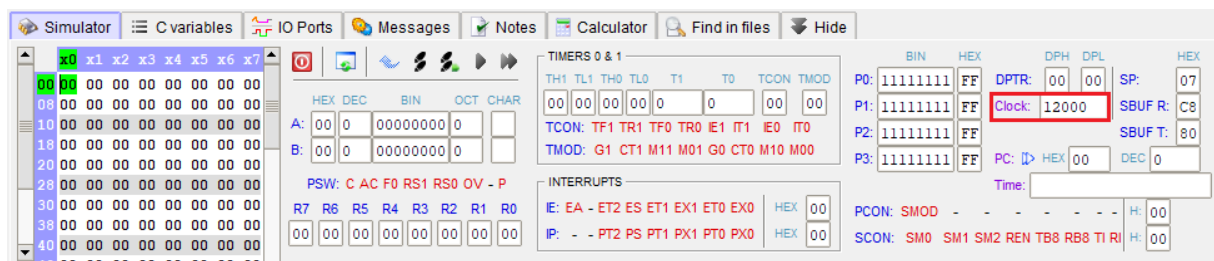
Symulacja wyświetlacza LCD w środowisku MCU 8051 IDE oddaje tą właściwość sterownika HD44780, jednak każde wejście do trybu symulacji bądź reset symulowanego mikrokontrolera przez naciśnięcie klawisza z ikoną  traktowane jest tak, jak włączenie zasilania dla całego systemu, czyli również symulowanego wyświetlacza LCD. Z tego powodu, przed wywołaniem funkcji realizującej w tworzonym programie inicjalizację sterownika HD44780 należy wywołać funkcję, która zrealizuje opóźnienie rzędu 10 ms. Napisanie takiej funkcji w języku C jest dość trudne, ponieważ podczas

pisania kodu nie wiemy, na jakie rozkazy mikrokontrolera zostanie zamieniony ten kod programu. Zatem nie jesteśmy też w stanie dokładnie określić czasu opóźnienia realizowanego przez taką funkcję. Łatwo to jednak zrobić definiując ciało funkcji za pomocą wstawki asemblerowej.

Na listingu 2 przedstawiono deklaracje takiej funkcji. Zamieszczone tam czasy wykonania poszczególnych rozkazów obliczono przy założeniu, że częstotliwość sygnału zegarowego doprowadzonego do mikrokontrolera wynosi 12 MHz. Zatem należy pamiętać, aby w przypadku wykorzystania tej funkcji w pisany programie w polu *Clock* na zakładce *Simulator* wpisać wartość 12000 (patrz rysunek 12), gdyż częstotliwość w tym polu jest wyrażana w kHz.

Listing 2. Funkcja realizująca opóźnienie o czasie trwania 10 ms

```
void LcdDelay10ms (void)
{
  __asm
  MOV R6,#20      ;opóźnienie o czasie 500 us zostanie wykonane 20 razy
00001$:
  MOV R7,#248    ;1us      \
00002$:         ;          |
  DJNZ R7,00002$ ;248*2us=496us | 500 us
  NOP            ;1us      |
  DJNZ R6,00001$ ;2us      /
  __endasm;
}
```



Rys.12. Widok zakładki *Simulator* z zaznaczonym polem *Clock*

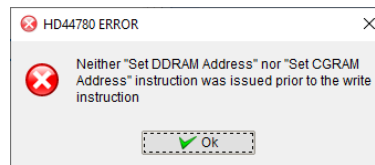
Przy pisaniu kodu funkcji realizującej odczyt informacji ze sterownika wyświetlacza należy pamiętać, aby nie odczytywać stanu linii danych bezpośrednio po zmianie linii E w stan wysoki, gdyż symulator wyświetlacza nie wystawi jeszcze na nich danych. Po ustawieniu linii E w stan wysoki, a przed odczytaniem informacji z linii danych sterownika powinno być zrealizowane opóźnienie o czasie nie krótszym niż 1 μ s. Można to łatwo zrealizować przez wstawkę asemblerową pokazaną na listingu 3.

Listing 3. Wstawka asemblerowa realizująca opóźnienie o czasie trwania 1 μ s

```
__asm
NOP      ;1us
__endasm;
```

Kolejną ważną rzeczą, jaką należy wziąć pod uwagę przy tworzeniu programu obsługi wyświetlacza LCD ze sterownikiem HD44780 uruchamianego w środowisku MCU 8051 IDE jest konieczność wysłania podczas jego inicjalizacji rozkazu ustawienia adresu w pamięci DDRAM. Bez

tego podczas pierwszej próby wyświetlenia znaku pojawi się okno z komunikatem jak na rysunku 13, a w oknie przedstawionym na rysunku 10 pojawi się błąd jak na rysunku 14.




Rys.13. Komunikat ostrzegający o baku ustawienia adresu w pamięci DDRAM sterownika wyświetlacza

```
[ERROR] Neither "Set DDRAM Address" nor "Set CGRAM Address" instruction was issued prior to the write instruction
```

Rys.14. Komunikat błędu w oknie *HD44780 log*

Aby oddać rzeczywiste zachowanie się wyświetlacza w rzeczywistych warunkach pracy, wspomniany rozkaz powinien ustawiać adres 00h.

Warto również pamiętać, aby po każdym ponownym przejściu do trybu symulacji resetować wirtualny model wyświetlacza. W tym celu oknie symulacji wyświetlacza LCD należy kliknąć klawisz z ikoną  i z rozwiniętego menu wybrać pozycję *Reset HD44780*.

Komentowanie kodu programu

Komentarze umieszczane w kodzie programu powinny opisywać działanie danego rozkazu lub grupy rozkazów z punktu widzenia funkcji, jakie pełnią w tworzonym programie. Dla przykładu, jeśli w kodzie stworzonego programu zaprezentowana poniżej linia kodu ma za zadanie ustawienia linii P1.1 mikrokontrolera w celu ustawienia stanu wysokiego na linii E sterownika wyświetlacza w stan wysoki, to komentarz powinien być następujący

```
// Ustawienie linii E w stan wysoki
```

```
P1_1 = 1;
```

a nie taki

```
// Ustawienie linii P1.1 w stan wysoki
```

```
P1_1 = 1;
```

Jego niepoprawność polega na tym, że nie daje on żadnej informacji odnośnie funkcji, jaką dany rozkaz pełni w tworzonym programie, a tylko powiela informację prezentowaną przez jego zapis.