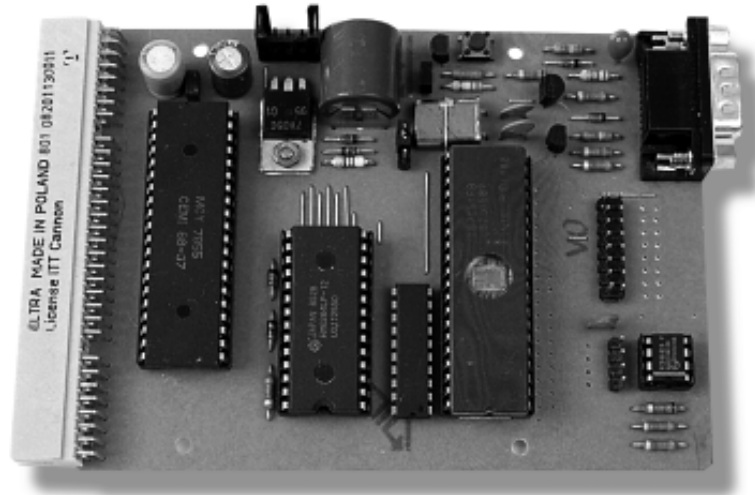


# Mikroprocesorowy system edukacyjny, część 2

## kit AVT-353

Drugą część artykułu rozpoczynamy od krótkiego opisu sposobu programowania portu I/O, stanowiącego „okno na świat” prezentowanego systemu mikroprocesorowego. W dalszej części artykułu omówiono programowanie wyświetlacza LCD z wbudowanym sterownikiem HD44780.



### Programowanie portu I/O

Opis systemu uruchomionego byłby niekompletny bez, chociaż skrótowego, omówienia sposobu programowania uniwersalnego układu wejścia/wyjścia U3. Jak wspomniano, udostępnia on trzy porty oznaczone jako PA, PB i PC. Porty PA i PB są niepodzielne, natomiast port PC jest podzielony logicznie na dwie części: mniej znaczącą tetradę PCl z bitami od PC0 do PC3, oraz bardziej znaczącą PCh (bity od PC4 do PC7). Obie części portu PC, jak i PA oraz PB, zaprogramować można jako niezależne wejścia, lub wyjścia. W zewnętrznej przestrzeni danych mikrokontrolera układ zajmuje cztery komórki pamięci. Przyjęto dla portu PA komórkę o adresie 0, dla PB adres 1, a dla PC adres 2. Ostatnią komórkę, oznaczoną adresem 3 zajmuje rejestr kontrolny CTRL. Zapisując do niego bajt danych uzyskany według poniższego wzoru możemy określić funkcję portów układu:

```
7 6 5 4 3 2 1 0
1 0 0 PA PCh 0 PB PCl
```

gdzie PA, PB, PCl, PCh, przy wartości 0 funkcjonuje jako wyjście, a przy 1 jako wejście.

Tak więc wpisując do rejestru CTRL (adres 3) wartość np. 8Ah

definiujemy porty PA oraz PCl jako wyjście, natomiast porty PB wraz z PCh jako wejście.

Przez wpisanie do rejestru CTRL inaczej utworzonego bajtu uzyskujemy możliwość indywidualnego ustawiania lub kasowania bitów portu PC zaprogramowanych uprzednio jako wyjścia:

```
7 6 5 4 3 2 1 0
0 0 0 0 PCn PCn PCn C/S
```

gdzie: PCn jest określonym dwójkowo numerem bitu portu (0..7), C/S o wartości 0 oznacza zerowanie wybranego bitu, a 1 jego ustawienie.

Wpisując do rejestru CTRL wartość np. 5, uzyskujemy ustawienie linii PC2 portu (zaprogramowanego uprzednio jako wyjście).

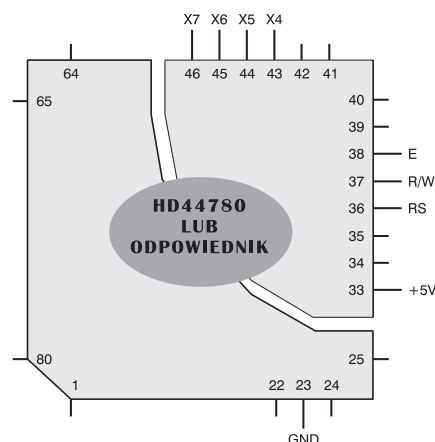
Ten krótki opis nie wyczerpuje wszystkich możliwości zastosowanego układu, jest jednak wystarczający dla rozpoczęcia pierwszych prób. Celem uzyskania szerszych informacji należy sięgnąć do kart katalogowych układu scalonego 8255 firmy INTEL.

### Adaptacja i programowanie wyświetlacza LCD

W kolejnej części opisu przedstawiony zostanie sposób przystosowania wyświetlacza ciekłokrystalicznego do systemu edukacyjnego oraz jego możliwości. Dołą-

czenie wyświetlacza otwiera szerokie możliwości prezentacji wyników pracy programów użytkownika. Wyświetlacz umożliwia, między innymi, wyświetlenie wszystkich znaków standardowego zbioru ASCII oraz dzięki wbudowanej w sterownik „inteligencji“ przeniesienie części zadań związanych z obsługą procesu wyświetlania bezpośrednio z programu użytkownika do kontrolera wyświetlacza. Dalszy opis dotyczy wyświetlacza alfanumerycznego o dowolnej organizacji (np. 1 linia po 16 znaków, 1x20, 2x16, itd.), sterowanego przez zintegrowany kontroler typu HD44780 firmy HITACHI lub jego odpowiednik.

Przede wszystkim należy ustalić kolejność wyprowadzeń wyświetlacza korzystając z rys. 5, który przedstawia położenie końcówek komunikacyjnych kontrolera, odpowiadających końcówkom dwóch złączy wykonanych na płycie systemu uruchomieniowego. Jeżeli odnaleziono w ten sposób wyprowadzenia wyświetlacza, odpowiadają bezpośrednio końcówkom złącza LCD1 lub LCD2, to wyświetlacz można podłączyć bezpośrednio. Jeżeli jednak producent zastosował inne rozwiązanie wyprowadzeń wyświetlacza, powstaje konieczność własnoręcznego wykonania przejściówki pomiędzy nim, a systemem uruchomieniowym. Należy zwrócić uwagę na zgodność oznaczeń wyprowadzeń scalonego kontrolera wyświetlacza z oznaczeniami występującymi w opisie końcówek złączy, wykonanych na płycie systemu (patrz schemat elektryczny). Dalszy opis będzie dotyczył programowania funkcji kontrolera.



Rys. 5. Wyprowadzenia sterownika Hitachi.

Do poznania możliwości kontrolera wyświetlacza użyjemy interfejsu równoległego komputera PC, oraz znajdującego się na dołączonej do systemu dyskietce programu *lcd4.exe*.

Aby rozpocząć test, wyświetlacz należy podłączyć do komputera PC w sposób przedstawiony na rys. 6. Pomiędzy złączem interfejsu a panelem zastosowano sześć rezystorów zabezpieczających przed ewentualnymi zwarzeniami, mogącymi powstać w trakcie późniejszych doświadczeń. Jeżeli zastosowany panel LCD nie posiada potencjometru regulacji kontrastu, to należy odnaleźć końcówkę 30 kontrolera na złączu panela. Umożliwia ona zewnętrzną regulację kontrastu - dla uproszczenia końcówkę tę należy tymczasowo podłączyć do masy.

Uruchomienie program testu możliwości kontrolera realizuje komenda : `<lcd4.exe [LPTn]>`, gdzie `<LPTn>` określa interfejs równoległy, przez który odbywać się będzie komunikacja z wyświetlaczem. Dopuszczalnymi nazwami są: LPT1, LPT2 oraz HGC dla interfejsu równoległego karty HERCULES lub części komputerów typu „notebook“. W razie pominięcia oznaczenia przyjęte zostanie domyślne złącze równoległe oznaczone LPT1.

Wszelkie operacje wykonywane z poziomu programu odbywają się w sposób analogiczny jak w systemie operacyjnym i polegają na wprowadzaniu z klawiatury komputera ciągów prostych poleceń. Tak więc doświadczenia zdobyte podczas prób przenoszą się w sposób bezpośredni do programów użytkownika (napisanych z użyciem procedur systemu operacyjnego). Po uruchomieniu program zgłasza się nagłówkiem mówiącym o oznaczeniu wybranego łącza równoległego oraz sposobie identyfikacji rozkazów i danych, po którym wyświetlany jest znak zachęty. Informuje on o możliwości wpisania ciągu znaków, który następnie zostanie przesłany do kontrolera wyświetlacza. Rozpoznawane są następujące sekwencje rozkazów i danych:

! - inicjacja sterownika panela, przełączenie w tryb komunikacji czterobitowej, ustawienie obsługi jednej linii o długości

80 znaków, np. bezpośrednio po uruchomieniu programu wprowadzić należy ciąg rozkazów: `!@0C@01` umożliwiający wyświetlenie tekstów wprowadzanych w dalszej kolejności z klawiatury;

@hh - dwuznakowa liczba heksadecymalna określająca rozkaz dla kontrolera;

#hh - dwuznakowa liczba heksadecymalna określająca daną przesłaną do pamięci kontrolera; sekwencja jest przeznaczona typowo dla definicji znaków użytkownika;

\$n - jednoznakowa liczba umożliwiająca wyświetlenie jednego z ośmiu znaków definiowanych przez użytkownika;

tekst- przesłanie ciągu znaków do pamięci kontrolera.

Znaki !, @, #, \$ są zastrzeżone i nie powinny być używane w innym kontekście, niż opisany wyżej. Próba niewłaściwego ich użycia jest sygnalizowana jako błąd. Pozostała część ciągu znaków, występująca po błędnym użyciu znaku zastrzeżonego jest ignorowana. Kontroler HD44780 realizuje następujące rozkazy, podzielone funkcjonalnie na trzy grupy:

1) Rozkazy o znaczeniu trwałym - po wprowadzeniu określone działanie utrzymuje się aż do wydania innego rozkazu z danej podgrupy:

- podgrupa A - organizacja wyświetlacza:

@20 - tryb jednoliniowy (stan po inicjacji);

@28 - tryb dwuliniowy;

- podgrupa B - kontrola wyświetlania:

@08 - wygaszenie wyświetlacza (stan po inicjacji);

@0C - włączenie wyświetlacza bez załączenia kursora;

@0D - włączenie wyświetlacza oraz załączenie kursora funkcjonującego jako pulsowanie znaku;

@0E - włączenie wyświetlania, oraz załączenie kursora funkcjonującego jako podkreślenie znaku (zależy od typu wyświetlacza);

- podgrupa C - sposób wprowadzania znaków:

@04 - kursor przesuwa się od pozycji początkowej w lewo, np. `@01@87@04<tekst>`, po czym przeskakuje na pozycję

końcówką linii znikającą z wyświetlacza - aby zobaczyć wprowadzone znaki należy kilkakrotnie powtórzyć rozkaz @1C, co zobrazuje na wyświetlaczu jednocześnie koniec (@CF) i początek linii (@80);

@05 - wstawia znak na pozycję kursora, przesuwa tekst w prawo, kursor pozostaje przy tym nieruchomy, np. @01@87@05<tekst> od lewej strony wyświetlacza pojawiają się końcowe znaki linii (ostatni znak - @CF, pierwszy - @80);

@06 - kursor przesuwa się od pozycji początkowej w prawo (stan po inicjacji), po czym znika z wyświetlacza - aby zobaczyć wprowadzone znaki należy kilkakrotnie powtórzyć rozkaz @18, co zobrazuje je na wyświetlaczu, jednocześnie jednak z lewej strony zniknie początek linii;

@07 - wstawia znak na pozycję kursora, przesuwa tekst w lewo, kursor pozostaje przy tym nieruchomy, np. @01@87@07<tekst>.

2) Rozkazy o znaczeniu chwilowym - po wykonaniu nie wpływają na dalsze działanie kontrolera panela:

- inicjacja pamięci wyświetlanych tekstów:

@01 - kasowanie wyświetlacza, kursor i okno wyświetlacza od pozycji początkowej linii (pozycja @80);

@02 - kursor i okno wyświetlacza od pozycji początkowej linii (pozycja @80);

- kierunek ruchu kursora lub wyświetlanego tekstu:

@10 - przesuwa kursor w lewo o znak; @14 - przesuwa kursor w prawo o znak;

@18 - przesuwa kursor wraz z wyświetlanym tekstem w lewo, odsłaniając kolejne znaki z prawej strony;

@1C - przesuwa kursor wraz z wyświetlanym tekstem w prawo, odsłaniając kolejne znaki z lewej strony;

- ustawienie pozycji kursora:

@<80h+nn> - dwuznakowa liczba heksadecymalna powstała w wyniku zsumowania bazy (80h), i przesunięcia (nn), gdzie:

<nn> - liczba heksadecymalna określająca położenie kursora w pamięci wyświetlanych tekstów; dla trybu jednoliniowego są dostępne pozycje kursora z zakresu: nn = 0..4F (80 znaków), wybierane rozkazami @80 do @CF, natomiast dla trybu dwuliniowego są dostępne pozycje kursora:

- pierwsza linia: nn = 0..27 (40 znaków), wybierane @80 do @A7;

- druga linia: nn = 40..67 (40 znaków), wybierane @C0 do @E7; np. umieszczenie kursora w ósmej pozycji realizuje rozkaz @87;

3) Rozkaz definicji znaków użytkownika:

@<01zzzwww> - dwuznakowa liczba heksadecymalna zapisana binarnie, określająca dostęp do programowanego wiersza wybranego znaku, gdzie:

<www> - numer wiersza znaku (0 do 6);

<zzz> - numer programowanego znaku (0 do 7).

Wszystkie bajty danych wprowadzane po tym rozkazie są kierowane do pamięci znaków definiowanych przez użytkownika - opuszczenie opcji możliwe jest po wydaniu rozkazu inicjacji pamięci lub ustawienia pozycji kursora (np.@02, lub @80), np. przykładowa definicja znaku zapisanego binarnie: 01110 - 0Eh, 00001 - 01h, 01111 - 0Fh, 10001 - 11h, 01111 - 0Fh, 00010 - 02h, 00001 - 01h, 00000 - 00h wygląda następująco:

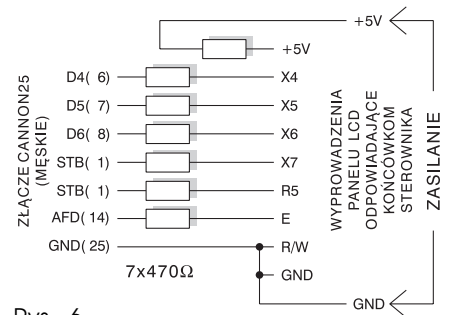
@01\$3 - wyświetlenie znaku niezdefiniowanego;

@58#0E#01#0F#11#0F#02#01#00 - definicja znaku;

@81 - opuszczenie opcji definicji znaku;

@01bl\$3d - wyświetlenie tekstu z zdefiniowanym znakiem.

Opis sposobu programowania kontrolera wyświetlacza wydawać się może enigmatyczny, jednak ze względu na występującą dużą różnorodność typów, różniących się przede wszystkim organizacją, trudno byłoby w sposób zwięzły przedstawić wszystkie aspekty ich programowania. Z konieczności ograniczono się do organizacji jednoliniowej, wychodząc z założenia, że organizacja dwuliniowa w niewielkim stopniu odbiega od przedstawionej w opisie. Zasadni-



Rys. 6.

cza różnica dotyczy właściwie jednego rozkazu - ustawienia pozycji kursora. Pamiętać jednak należy, aby bezpośrednio po uruchomieniu programu lcd4.exe zainicjować kontroler dla trybu dwuliniowego ciągiem rozkazów: !@28@0C@01.

Złożony wydawać się może sposób opisu niektórych rozkazów - praktyka autora pokazuje, że dla efektywnego użycia panela LCD, jest konieczna znajomość rozkazów z grupy drugiej (rozkazy o znaczeniu chwilowym). Rozkazy z grupy pierwszej mają znaczenie raczej drugoplanowe, natomiast rozkaz z grupy trzeciej w praktyce realizowany jest przez specjalizowaną procedurę systemu operacyjnego.

Celem uzupełnienia należy dodać, że sposób zestawienia kolejności rozkazów realizowanych przez kontroler wyświetlacza został podyktowany koniecznością maksymalnego uproszczenia oraz ujednolicenia opisu. Odbiega on tym samym od informacji, jakie znaleźć można w katalogach firmy HITACHI lub innych, produkujących odpowiednik opisanego sterownika.

**Krzysztof Kuryłowicz**