

Procedury obsługi wyświetlaczy LCD, część 1

Ogromne zainteresowanie, jakim cieszyły się artykuły poświęcone sterowaniu alfanumerycznych i graficznych wyświetlaczy LCD, zachęciło nas do kontynuowania tematu.

W pierwszej części artykułu przedstawiamy kompendium wiedzy o praktycznym stosowaniu alfanumerycznych wyświetlaczy LCD, łącznie z przykładowym programem obsługi i schematami ilustrującymi sposób ich dołączenia do mikrokontrolera.

Sposoby „komunikowania się” układów sterowanych mikroprocesorowo z użytkownikiem stale się zmieniają i są coraz bardziej doskonałe. Powoli „normalnymi” stają się układy mówiące do nas zapamiętanym lub zszyntetyzowanym głosem, wyświetlające rozmaite informacje w sposób graficzny za pomocą różnych obrazów. Nie maleje jednak popularność zwykłych, siedmiosegmentowych wskaźników i ich różnych odmian - głównym powodem jest bardzo atrakcyjna cena. Co jednak zrobić, gdy musimy wyświetlić informację w rodzaju: „za mało inteligentny, aby posługiwać się tym urządzeniem...”? W takim przypadku proponuję sięgnąć do wyświetlacza alfanumerycznego.

Wyświetlacz alfanumeryczny, najczęściej ciekłokrystaliczny lub gazowy, stał się na tyle popularnym, że nie trzeba nikogo przekonywać o zaletach jego stosowania. Dostyc jest powiedzieć, że daje bardzo szerokie możliwości obrazowania informacji, począwszy od typowych znaków ASCII aż do definiowanych przez użytkownika, czyli prawie dowolnych. W odróżnieniu od wyświetlacza graficznego opisywanego w EP 6-7/99 umożliwia jednak wyświetlanie łącznie znaków, których definicje umieszczono wcześniej w wewnętrznej pamięci, tak zwanym generatorze znaków.

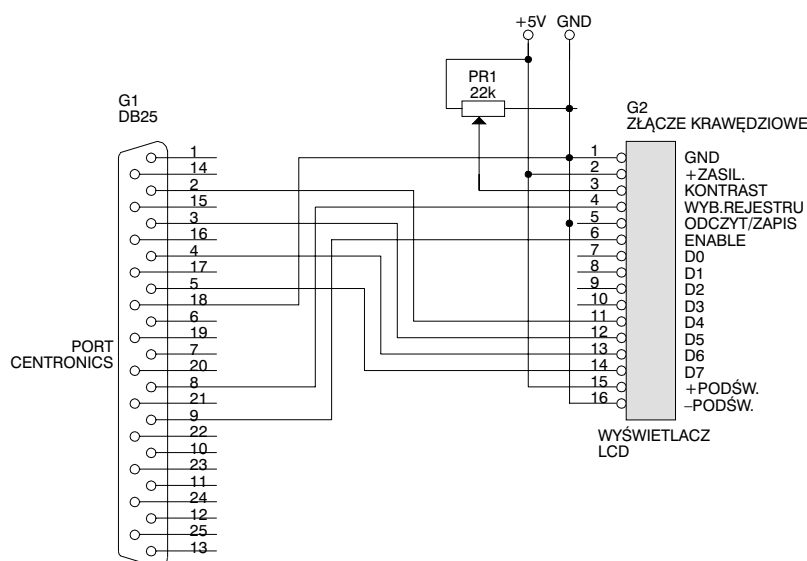
Celem artykułu jest opisanie propozycji podłączenia wyświetlacza znakowego do systemu mikroprocesorowego opartego o rodzinę MCS51. W opisywanych modelach użyłem wyświetlacza PVC200403-PYL01 i do niego to odnoszą się wszystkie aplikacje. Wybór wyświetlacza podyktowany był przystępną ceną przy dobrych walorach użytkowych (podświetlanie na diodach LED, wyświetlane 4 linie po 20 znaków). Prezentowane aplikacje można odnieść do dowol-

nego wyświetlacza alfanumerycznego LCD, ponieważ większość producentów używa tego samego algorytmu ich sterowania oraz tego samego zestawu rozkazów. Mało tego, w większości możemy znaleźć taki sam układ sterownika, to jest HD44780 (nota aplikacyjna dostępna w Internecie pod adresem: <http://www.ep.com.pl/ftp/hd44780.pdf> oraz na płycie CD-EP02/2000 w katalogu \Noty katalogowe do projektów\LCD).

Trochę teorii

W budowie wyświetlacza możemy wyróżnić cztery podstawowe elementy: pamięci ROM i RAM, kontroler obsługujący wyświetlanie oraz ekran ciekłokrystaliczny.

Pamięć RAM to tak zwana pamięć obrazu. Jej nazwa bierze się stąd, że ściślijm jej odwzorowaniem jest obraz na ekranie wyświetlacza. W odróżnieniu od pamięci obrazu w wyświetlaczu graficznym, zapamiętywane są w niej nie pojedyncze bity obrazu, a kody znaków w generatorze znaków, czyli pamięci ROM lub RAM zawierającej kody opisujące wygląd (matryce) znaku. Sterownik wyświetlający obraz pobiera z pamięci obrazu kod znaku z generatora znaków i punkt po punkcie odwzorowuje jego wygląd na ekranie LCD. Stąd też aplikacja porozumiewa się z wyświetlaczem za pomocą kodów znaków, a nie przesyłając pojedyncze bity obrazu. Zazwyczaj część lub całość generatora znaków przepisywana jest podczas startu wyświetlacza z pamięci ROM do RAM i dzięki temu mamy możliwość



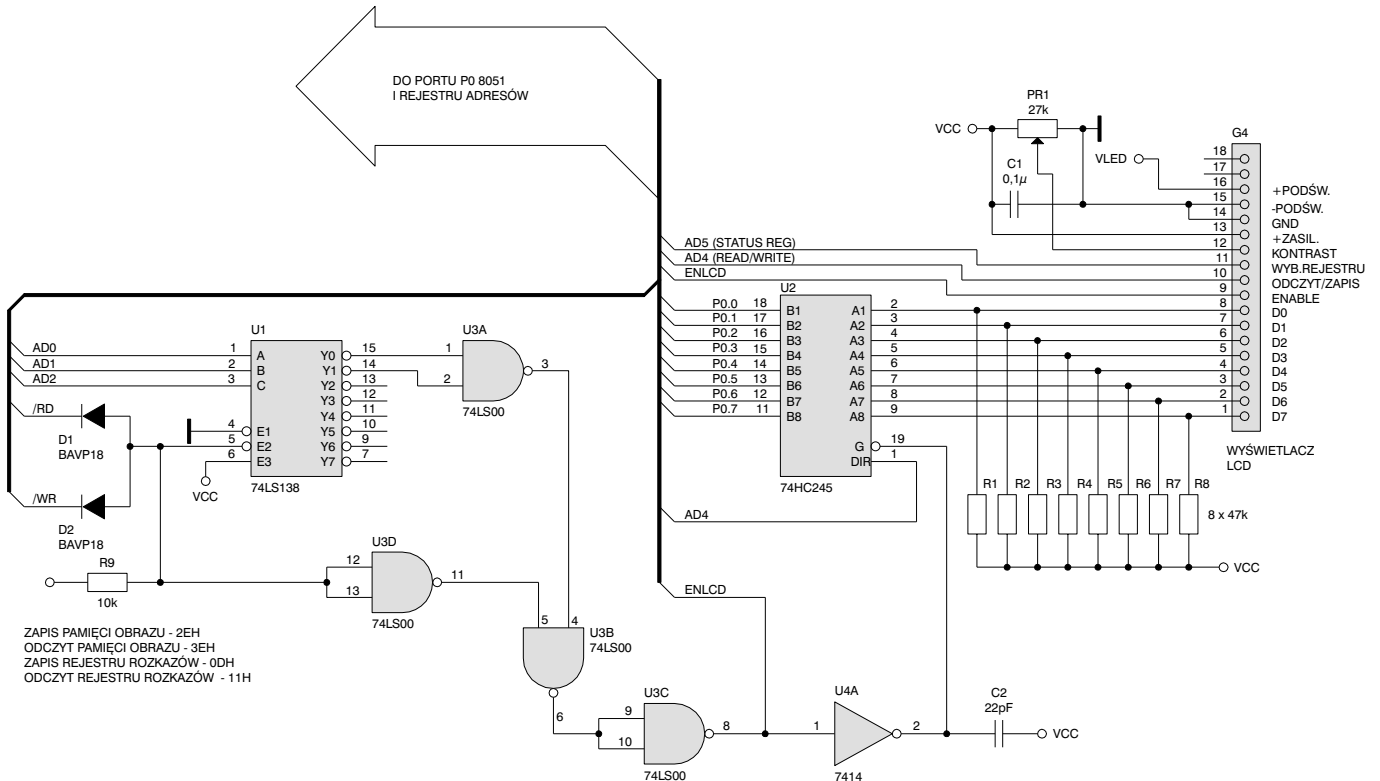
Rys. 1.

Tab. 1. Sposób zapisania definicji znaku w generatorze znaków na przykładzie definicji znaku większości.

Nr bajtu definicji	7	6	5	4	3	2	1	0	Wartość hex
0	X	X	X	00
1	X	X	X	00
2	X	X	X	1	1	.	.	.	18
3	X	X	X	.	1	1	.	.	0C
4	X	X	X	.	.	1	1	.	06
5	X	X	X	.	1	1	.	.	0C
6	X	X	X	1	1	.	.	.	18
7	X	X	X	00

Tab. 2. Kombinacje sygnałów RS oraz R/W sterujące wyborem rejestru danych/rozkazy wyświetlacza.

Funkcja	Poziomy sygnał RS i R/W
Zapis do pamięci znaku lub jego wzorca	RS=H, R/W=L
Odczyt z pamięci znaku lub jego wzorca	RS=H, R/W=H
Zapis rozkazu	RS=L, R/W=L
Odczyt znacznika zajęty / wolny (BUSY / READY)	RS=L, R/W=H



Rys. 2.

definiowania - zmiany znaku. W takim przypadku znak definiowany zastępuje oryginalny, wykorzystując jego kod. Rodzaj pamięci zawierającej generator znaków ma bardzo istotne znaczenie praktyczne, ponieważ tylko wyświetlacze, które mają generator znaków w RAM (tzw. CG RAM) umożliwiają definiowanie własnych znaków. Dla kogoś, kto zmuszony będzie dorobić „polskie ogonki“ do liter może to mieć bardzo duże znaczenie.

Przy zapisie pamięci obrazu czy generatora znaków (zawartego w RAM!) obowiązują wszystkie zasady dotyczące zapisu pamięci, tzn. pamięci te mają swój czas dostępu i swoją przestrzeń adresową. Przed zapisem należy ustawić adres, pod który wysyłane będą dane i rodzaj zapisywanej pamięci (lub rejestru rozkazów), a aplikacja sterująca powinna badać stan wyświetlacza lub też sterować nim na tyle wolno, aby było dosyć czasu na realizację przez procesor wszystkich jego funkcji. W praktyce jest to czas najdłuższej wykonywanej przez kontroler wyświetlacza operacji i wynosi około 2 ms. W wyświetlaczu PVC zarówno generator znaków, jak i pamięć obrazu zaczynają się od adresu 0. Sposób zapisu wzorów znaków różni się tylko ustawieniem odpowiedniego trybu pracy wyświetlacza.

Wzór znaku ma postać matrycy 5x7 lub 5x10 punktów. Poziom H oznacza zaświecenie (w LCD zaczerpienie) punktu, a poziom L jego zgaszenie. W przypadku wyświetlacza w mojej aplikacji tryb 5x10 nie był zbyt użyteczny, toteż skupiłem się wyłącznie na trybie 5x7 i jego też dotyczą przykłady definicji własnych znaków. Sposób w jaki są zapamiętywane definiowane znaki przedstawiono w tab. 1.

Komunikacja z wyświetlaczem może odbywać się za pomocą interfejsu 4 lub 8-bitowego. W tym pierwszym przypadku należy wysłać najpierw bardziej znaczącą część bajtu rozkazu czy danych (bity b4 do b7), a później mniej znaczącą (bity b0 do b3). Oczywiście ten sposób komunikacji jest wolniejszy, lecz oszczędza nam 4 bity danego portu, a w praktyce nie zauważa się żadnej widocznej różnicy w czasie pracy.

Jedynym utrudnieniem jest komplikacja programu sterującego, który musi podzielić bajt i wysłać go w dwóch „porcjach“. Przy takim sterowaniu do przesyłania danych wykorzystuje się linie DB4 do DB7 wyświetlacza, a DB0 do DB3 zostawia się nie podłączone.

Dane podawane na wejście są zatrzaskiwane opadającym zboczem sygnału ENABLE. Istotne jest, aby różnica pomiędzy zmia-

Tab. 3. Wykaz rozkazów akceptowanych przez wyświetlacz z kontrolerem HD44870.

Nazwa	Kod	Opis funkcji
Clear display (czyść ekran)	01h	Kasuje obraz na ekranie i ustawia kursor w pozycji wyjściowej
Return home (powrót do pozycji 0,0)	02h	Przesuwa kursor do pozycji wyjściowej; również przywraca obraz gdy był on przesuwany rozkazem SHIFT
Entry mode set (ustawienie trybu zapisu danych)	07h do 04h	Ustawia kierunek ruchu kursora przy zapisie danych oraz decyduje o przesuwaniu obrazu (b0=L: nie przesuwać obrazu, b0=H: przesuwać obraz; b1=L: kursor w lewo, b1=H: kursor w prawo)
Display ON/OFF control (kontrola ekranu)	08h do 0Fh	Funkcje kontroli ekranu lcd (b0=L wyłącza migotanie kursora, b0=H załącza; b1=L wyłącza kursor, b1=H załącza; b2=L wyłącza ekran, b2=H załącza)
Cursor and display shift (przesuwanie kursora i obrazu)	10h do 1Ch	Przesuwanie kursora (b2=L przesuwanie kursora, b2=H przesuwanie całego obrazu; b3=L przesuwanie w prawo, b3=H przesuwanie w lewo; b1 i b0 nie mają znaczenia)
Function set (ustawienie funkcji)	20h do 2Ch	Ustawienie rejestru funkcji (b2=L znaki 5x7, b2=H znaki 5x10; b3=L wyświetlana jest 1 linia, b3=H wyświetlane są 2 linie; b4=L interfejs 4 bity, b4=H interfejs 8 bitów)
Set CG RAM address (ustawienie adresu CG RAM)	40h do 7Fh	Ustawienie adresu generatora znaków, pod który zapisywane będą definicje znaków lub spod którego będą czytane dane; adres podawany na liniach db5..0; również włączenie trybu zapisu do CG RAM
Set DD RAM address (ustawienie adresu pamięci obrazu)	80h do FFh	Ustawienie adresu pamięci obrazu, pod który zapisywane będą dane lub spod którego będą one odczytywane (może służyć do pozycjonowania kursora w miejscu znaku); adres podawany na liniach db6..0; również przełączenie z trybu definicji do wyświetlania
Read busy flag & address (odczyt stanu zajętości oraz adresu)	80h i R/W=1	Odczyt znacznika stanu zajętości oraz aktualnego adresu wskazywanego przez licznik adresów; b7=L wyświetlacz zajęty, b7=L wyświetlacz gotowy; pozostałe bity przekazują aktualny adres licznika adresów

List. 1. Opis źródłowy dekodera implementowanego w układzie GAL16V8 (rys. 3) napisany w języku PALASM.

```

TITLE          Dekoder wyświetlacza
PATTERN       1
REVISION      0
AUTHOR        Jack
COMPANY       EP
DATE          07/15/99
CHIP          _dek    PALCE16V8

;----- Deklaracje wyprowadzeń -----
PIN 5         /RD          COMBINATORIAL ; INPUT
PIN 6         /WR          COMBINATORIAL ; INPUT
PIN 7         LE           COMBINATORIAL ; INPUT
PIN 8         SE0         COMBINATORIAL ; INPUT
PIN 9         SE1         COMBINATORIAL ; INPUT
PIN 10        GND
PIN 11        /OEN        COMBINATORIAL ; INPUT
PIN 14        RDLCD       COMBINATORIAL ; OUTPUT
PIN 15        DALCD       COMBINATORIAL ; OUTPUT
PIN 16        ENLCD       COMBINATORIAL ; OUTPUT
PIN 20        VCC

;----- Funkcje boolowskie ---
EQUATIONS
MINIMIZE_OFF
DALCD=/LE*DALCD+LE*SE0+SE0*DALCD
RDLCD=/LE*RDLCD+LE*SE1+SE1*RDLCD
MINIMIZE_ON
ENLCD=RDLCD*RD+RDLCD*WR

MOORE_MACHINE

S0 := VCC -> S7
S1 := VCC -> S6
S2 := VCC -> S5
S3 := VCC -> S4
S4 := VCC -> S3
S5 := VCC -> S2
S6 := VCC -> S1
S7 := VCC -> S0

```

na ENABLE a zanikiem ważnych danych na szynie była większa niż 10 ns - nie może to być proces jednoczesny. Normalnie, gdy nie są transmitowane żadne dane, sygnał ENABLE powinien mieć poziom L. Przed jego zmianą na H powinny być ustabilizowane wartości sygnałów odczyt/zapis (R/W) oraz wybór rejestru (RS). W praktyce również okazało się, że minimalny czas trwania sygnału ENABLE, podawany przez producenta jako 450 ns, nie ma znaczącego wpływu na pracę wyświetlacza. Można go skró-

cić nawet do 200 ns. Kombinacje wartości sygnałów sterujących wyświetlaczem przedstawiono w **tab. 2**.

Warunkiem poprawnego funkcjonowania wyświetlacza jest właściwe ustawienie jego parametrów zaraz po włączeniu i ustabilizowaniu się napięcia zasilania. Sekwencja inicjująca ustawia długość słowa danych oraz ustala inne ważne parametry pracy. Czasami specyfikacja właściwości wyświetlacza określa rodzaj standardowej inicjacji wykonywanej przez jego kontroler po za-

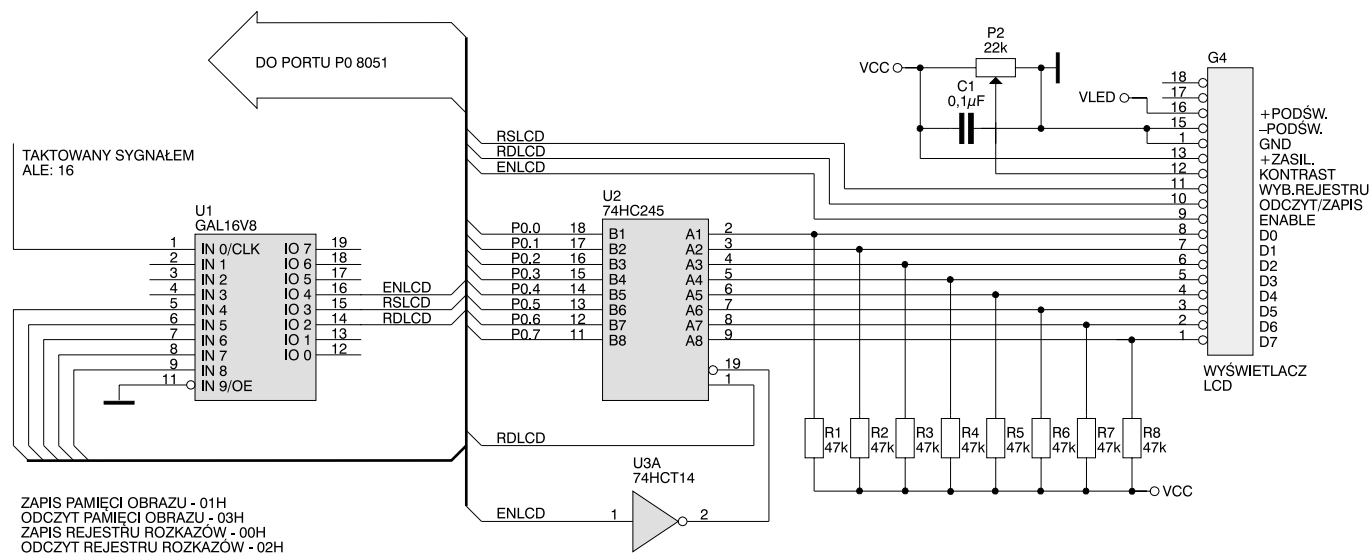
łączeniu napięcia zasilania, czasami musimy przeprowadzić ją sami. Zanim jednak zajmiemy się inicjowaniem wyświetlacza i jego trybów, to musimy wspomnieć o rozkazach akceptowanych przez sterownik wyświetlacza. Jest to lista kilku podstawowych, lecz bardzo użytecznych funkcji, które może wykonać sterownik niejako sam z siebie zwalniając nas z obowiązku kasowania pamięci obrazu czy też wyliczania współrzędnych powrotu do pozycji początkowej kursora (0,0). Oprócz takich typowych „użytków“ znajdują się wśród nich również inne, mniej praktyczne, ustawiające po prostu tryb pracy. Listę rozkazów wyświetlacza ze sterownikiem HD44870 ilustruje **tab. 3**.

Coś dla praktyków

Proponuję wykonanie bardzo prostego układu do uruchomienia wyświetlacza i testowania jego funkcji, wymagającego tylko złącza do portu Centronics komputera PC lub innego, złącza krawędziowego nadającego się do podłączenia wyświetlacza (albo po prostu jego przylutowanie), zasilacza napięcia stabilizowanego +5V oraz kilku przewodów. Ja użyłem typowego kabla do drukarki, któremu obciąłem złącze drukarkowe i przylutowałem krawędziowe, wylutowane zresztą ze starej płyty głównej. Schemat połączeń ilustruje **rys. 1**.

W proponowanym układzie dane przesyłane są przez interfejs 4-bitowy. Transmisja jest jednokierunkowa (tylko zapis), sygnał R/W zwarty jest na stałe do masy. Flaga zajętości wyświetlacza nie jest badana. Instrukcja *delay* określa opóźnienie pomiędzy operacjami zapisu danych i daje sterownikowi wyświetlacza dosyć czasu na realizację dowolnej funkcji.

Program obsługi napisany został w języku Turbo Pascal 7.0 dla DOS (kod źródłowy dostępny w Internecie pod adresem: http://www.ep.com.pl/ftp/lcd_prakt.exe oraz na płycie CD-EP2/2000 w katalogu \Noty katalogowe do projektów\LCD). Prezentowany układ ma tę zaletę, że można niewielkim kosztem opracować sobie algorytmy obsługi



Rys. 3.

wyświetlacza, zanim sięgniemy do języka assembler i do rozwiązań sprzętowych. Może również służyć do przeprowadzenia testów, ponieważ działa z całą pewnością (dodatkowo Pascal czy C mają świetny debugger) i może rozwiązać nasze wątpliwości co do tego, czy wyświetlacz funkcjonuje, czy też my popełniamy jakiś błąd przy jego obsłudze.

Przy okazji eksperymentów z wyświetlaczem proszę zwrócić uwagę na to, że w przypadku zapisu znaku od adresu 0 do 80 znaki wyświetlane są w kolejności linii: 1, 3, 2, 4. Program obsługi nie może więc wpisywać znaków wprost, lecz musi uwzględniać ten przeplot. Również instrukcje przesuwające zawartość obrazu zachowują się nieco „dziwnie“ w momencie, gdy znaki przekraczają krawędź obrazu i są przenieszone na drugą stronę. Zachęcam do eksperymentów!

Program sterujący rozpoczyna procedurę inicjowania wyświetlacza. Niektórzy producenci zalecają przeprowadzenie inicjacji tylko wówczas, gdy nie jest wystarczająca ta, którą sterownik wyświetlacza przeprowadza po włączeniu zasilania. Jednak z moich doświadczeń wynika, że dobrze jest mimo wszystko odzależnić te kilkanaście milisekund po włączeniu i zainicjować wyświetlacz. Czasami zdarza się bowiem tak, że z siebie tylko wiadomych powodów sterownik nie wykonuje inicjacji i nie pozwala zapisywać do wyświetlacza żadnych danych. Algorytmy inicjacji dla interfejsu 4- i 8-bitowego ilustruje **tab. 4**.

Na **rys. 2 i 3** pokazano schematy podłączenia wyświetlacza do systemów z mikroprocesorem z rodziny MCS51. W modelu używałem AT89S8252 z zewnętrznym rezonatorem 16 MHz i 4 MHz. Schemat ideowy przedstawiony na **rys. 2** wykorzystuje układy serii TTL-LS i TTL-HC. Wyjście danych do wyświetlacza buforowane jest za pomocą układu dwukierunkowego bufora danych 74HCT245. Diody D1 i D2 pracują w ukła-

Tab. 4. Wymagane sekwencje inicjalizacji wyświetlacza. Flaga zajętości może być sprawdzana po wykonaniu kroku 9.

Interfejs 8 bitów	Interfejs 4 bity
1. Załączenie napięcia zasilania	1. Załączenie napięcia zasilania
2. Czekaj nie mniej niż 15 ms po osiągnięciu przez napięcie zasilania wartości 4,5V	2. Czekaj nie mniej niż 15 ms po osiągnięciu przez napięcie zasilania wartości 4,5V
3. RS=L,R/W=L,dane=30h	3. RS=0,R/W=0,dane=3h
4. Czekaj nie mniej niż 4,1 ms	4. Czekaj nie mniej niż 4,1 ms
5. RS=L,R/W=L,dane=30h	5. RS=L,R/W=L,dane=3h
6. Czekaj nie mniej niż 100μs	6. Czekaj nie mniej niż 100 μs
7. RS=L,R/W=L,dane=30h	7. RS=L,R/W=L,dane=3h
8. Czekaj nie mniej niż 100μs	8. Czekaj nie mniej niż 100 μs
9. RS=L,RW=L,dane=Function set	9. RS=L,R/W=L,dane=2h
10. RS=L,R/W=L,dane=08h	10. R/S=L,R/W=L,dane=4 b.zn.bity Function set
11. RS=L,R/W=L,dane=01h	R/S=L,R/W=L,dane=4 mniej znaczące bity
12. RS=L,R/W=L,dane=Entry mode set	11. R/S=L,R/W=L,dane=0h
	R/S=L,R/W=L,dane=8h
	12. R/S=L,R/W=L,dane=0h
	R/S=L,R/W=L,dane=1h
	13. R/S=L,R/W=L,dane=0h
	R/S=L,R/W=L,dane=4 mniej znaczące bity
	Entry mode set

dzie bramki AND i dosyć dziwnym może się wydawać ich użycie, ale w układzie nie dysponowałem już żadną wolną bramką. Diody doskonale spełniają postawione przed nimi zadanie.

Rozwiązanie przedstawione na **rys. 3** wykorzystuje jako dekodery adresów układ GAL16V8. W projekcie sterowanie pracą wyświetlacza jest tylko jedną z funkcji układu GAL, ale równie dobrze może być jego podstawową w innym urządzeniu. Układ GAL zastępuje z powodzeniem co najmniej 2 układy scalone. Jego stosowanie ma jednak pewną wadę: trzeba dysponować programatorem i znać język opisu układów PLD. Na **list. 1** pokazano opis dekodera realizowanego w strukturze układu GAL16V8 uwzględniający część dotyczącą obsługi wyświetlacza.

Na jednym i drugim schemacie dekodery adresów zbudowany jest w taki sposób, aby rozkazy MOVX @Rn,A lub MOVX A,@Rn były wystarczające do współpracy z wyświetlaczem i nie jest wymagane żadne dodatkowe wypracowywanie przebiegów cza-

sowych. Oprócz sterowania wyświetlaczem LCD aplikacje mogą pełnić rolę dekodera adresów również dla klawiatury czy innych urządzeń zewnętrznych. Wszystko zależy od potrzeb.

Na zakończenie opisu wyświetlacza warto wspomnieć, że diody LED używane do podświetlenia tła ekranu LCD wymagają około 300 mA prądu przy napięciu rzędu 4V i warto jest zasilic je przed stabilizatorem tak, aby nadmiernie go nie przeciążać.

Jacek Bogusz, AVT
jacek.bogusz@ep.com.pl

Listingi programów omawianych w artykule dostępne są pod adresem http://www.ep.com.pl/ftp/lcd_prakt.exe oraz na płycie CD-EP02/2000 w katalogu \Noty katalogowe do projektów\LCD.

Nota katalogowa sterownika HD44780 dostępna jest pod adresem: <http://www.ep.com.pl/ftp/hd44780.pdf> oraz na płycie CD-EP2/2000 w katalogu \Noty katalogowe do projektów\LCD.