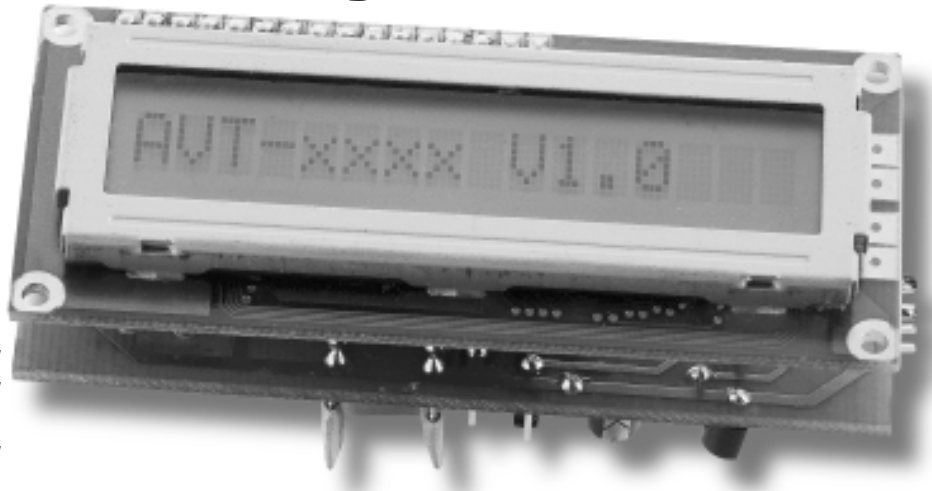


Szeregowy interfejs do wyświetlaczy LCD

AVT-5047



Sterowanie alfanumerycznym wyświetlaczem LCD jest dość kłopotliwe. Trzeba znać rozkazy sterujące pracą kontrolera wyświetlacza, dość kłopotliwa jest procedura inicjalizacji, występuje odwieczny problem z polskimi znakami, a do sterowania wyświetlaczem trzeba użyć minimum 7 linii procesora. W artykule przedstawiamy układ umożliwiający zmniejszenie liczby tych linii do jednej.

Wprawdzie linię RW można na stałe podłączyć do poziomu „L”, oszczędzając jedną linię procesora, ale takie sterowanie ma jednak poważną wadę - małą prędkość. Spowodowane jest to tym, że nie można stwierdzić, kiedy wyświetlacz zakończył wykonywać poprzednią komendę i przed wysłaniem następnej trzeba zawsze odczekać maksymalny czas katalogowy, który z reguły jest dużo dłuższy niż rzeczywisty czas wykonania komendy.

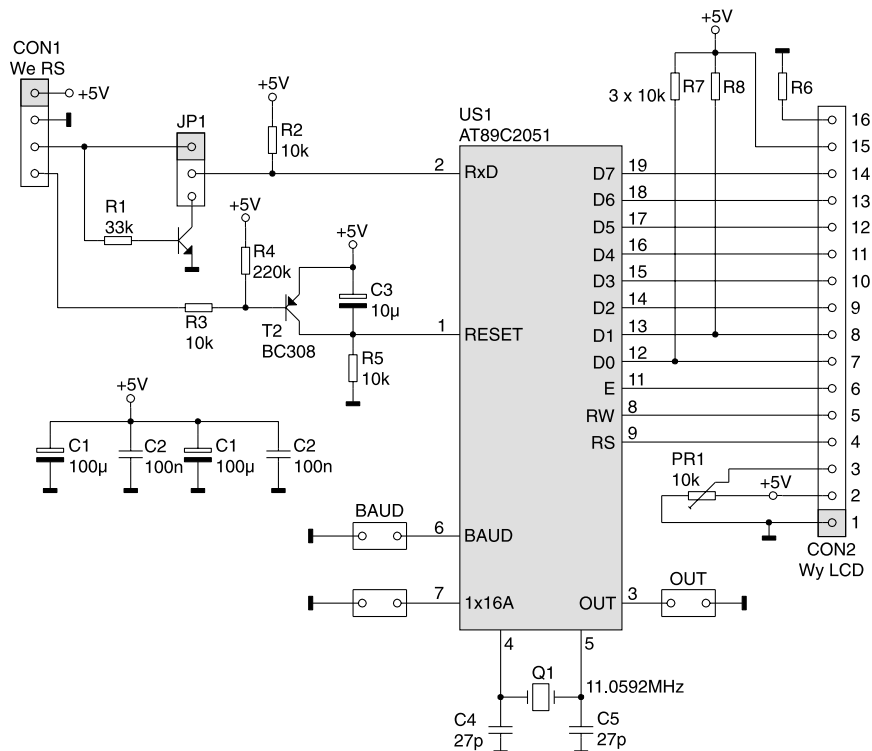
W artykule przedstawiamy szeregowy interfejs do wyświetlaczy LCD. Interfejs ten jest sterowany jedną linią procesora. Ponadto omijają nas kłopoty związane z inicjalizacją wyświetlacza, wpisaniem wzorów polskich znaków i nietypową obsługą wyświetlaczy 1x16 znaków. Układ interfejsu jest więc udoskonaloną wersją układów scalonych EDE700 i EDE702 opisanych w EP11/99, a przy tym zgodny z nimi programowo.

Budowa i zasada działania

Schemat interfejsu pokazano na rys. 1. „Sercem” układu jest mikrokontroler z wewnętrzną pamięcią programu. Po załączeniu zasilania procesor jest zerowany za pomocą obwodu różniczkującego składającego się z kondensatora C3 i rezystora R5. Układ można wyzerować podając napięcie o niskim poziomie logicznym na wyprowa-

dzenie 4 złącza CON1. Informacja podawana szeregowo w standardzie RS232 z wyprowadzenia 3 złącza CON1 jest kierowana do wejścia RxD procesora bezpośrednio (sygnał TTL) lub za pośrednictwem konwertera na tranzystorze T1 (sygnały zgodne z poziomami RS232C). Wyboru standardu sygnału dokonujemy jumperem JP1. Jeśli zwarte będą jego styki 1-2, to interfejs akceptuje poziomy TTL, jeśli natomiast zwarte będą styki 2-3 akceptowane będą poziomy RS232C. Najczęściej będziemy korzystać z pierwszej możliwości, dlatego można nie montować elementów R1 i T1.

Wyświetlacz LCD jest dołączony do złącza CON2. Istnieje możliwość włączenia podświetlenia wyświetlacza LCD (jeśli takie posiada). W tym celu należy dobrać wartość rezystancji R6. Zależy ona od typu podświetlenia LCD. Dla podświetlenia LED wartość R6 nie przekracza 22Ω. Potencjometrem PR1 regulujemy kontrast wyświetlacza. Zworką BAUD ustalamy prędkość transmisji. Dla zworki rozwartej ustawiana jest prędkość transmisji na 2400bd i format ramki 8N1, dla zwartej - prędkość 9600bd i format ramki 8N1. Zworkę 1x16A zakładamy, gdy zastosowaliśmy wyświetlacz 1x16 znaków. W rzeczywistości jest to wyświetlacz 2x8 znaków połączonych tak, aby tworzyły jedną linię. Wyświetlacz taki łatwo poznać po tym, że zawiera tylko jeden układ



WYKAZ ELEMENTÓW

- Rezystory**
 R1: 33kΩ
 R2, R3, R5...R8, PR1: 10kΩ
 R4: 220kΩ
- Kondensatory**
 C1: 100µF/16V
 C2: 100nF
 C3: 10µF/16V
 C4, C5: 27pF
- Półprzewodniki**
 T1: BC238 (BC547)
 T2: BC308 (BC557)
 US1: Zaprogramowany AT89C2051
- Różne**
 CON1: Listwa goldpin 4*1
 CON2: Gniazdo do listwy goldpin 16*1
 listwa 16*1 do wyświetlacza LCD
 JP1: Listwa goldpin 3*1 + jumper
 OUT: Listwa goldpin 2*1
 BAUD: Listwa goldpin 2*1 + jumper
 1x16A: Listwa goldpin 2*1 + jumper
 Q1: Kwarc 11.0592MHz

Rys. 1. Schemat elektryczny szeregowego interfejsu do wyświetlacza LCD.

scalony. W takim przypadku interfejs dodatkowo przesuwania kursora do drugiej linii po przekroczeniu 8 znaku. Dzięki temu na wyświetlaczu widoczne jest 16 znaków bez dodatkowych kombinacji z położeniem kursora. Wyjście OUT jest sterowane odpowiednimi rozkazami wysyłanymi przez RS232 i może służyć np. do włączenia podświetlenia.

Montaż i uruchomienie

Schemat montażowy płytki drukowanej interfejsu pokazano na rys. 2. Jego montaż rozpoczynamy od wlutowania zwojów, a następnie pozostałych elementów, od najmniejszych aż do największych. Złącze CON2 lutujemy od strony druku. Dzięki temu po wlutowaniu w wyświetlacz drugiej części złącza, całość można złożyć w „kanapkę”. Zworką 1x16A wybieramy typ wyświetlacza (1x16A/pozostałe). Po załączeniu zasilania na wyświetlaczu pojawią się komunikaty powitalne, a po chwili na wyczyszczonym ekranie migający kursor. Zworką JP1 ustalamy napięciowy standard sygnałów sterujących TTL/RS232C, a zworką BAUD prędkość transmisji.

Wstępnie uruchomiony interfejs można dołączyć do procesora lub do portu RS232C komputera. Na stronie internetowej EP (w dziale

Download>Dokumentacje) dostępne są skrypty demonstrujące możliwości wyświetlacza. Skrypty są dostępne dla komputera Amiga (demo.exe) i komputerów zgodnych z IBM PC (demo.bat).

Obsługa interfejsu

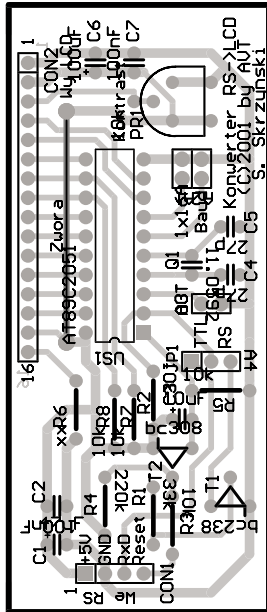
Do sterowania wyświetlaczem wystarczą w zasadzie podstawowe komendy akceptowane także przez wcześniej wspomniane układy z serii EDE. Wysyłając kody ASCII znaków do wyświetlania powodu-

jemy pojawianie się ich na wyświetlaczu; kursor będzie przesuwany w prawo. Wysłanie kodu sterującego Enter spowoduje przeniesienie kursora do drugiej linii wyświetlacza. Zastosowano także kilka innych kodów specjalnych, których zestawienie znajduje się w tab. 1.

Dzięki temu, że interfejs jest wyposażony w bufor wejściowy o wielkości 64 bajtów, po wykonaniu np. rozkazu czyszczenia wyświetlacza nie musimy zawieszać transmisji danych do czasu, aż wyświetlacz wykona tę operację. Przychodzące dane zapisane

Tab. 1. Znaczenie kodów specjalnych.

Wartość	Znaczenie
\$00	kod NULL (nie jest drukowany)
\$08	kod litery „ą”
\$01 lub \$09	kod litery „ę”
\$02	kod litery „ć”
\$03 lub \$0B	kod litery „ł”
\$04	kod litery „ń”
\$05	kod litery „ó”
\$06 lub \$0E	kod litery „ś”
\$07 lub \$0F	kod litery „ż” (ze względu na małą pamięć znaków nie ma litery „ź”)
\$0A	kod znaku LF przesuwania kursora do drugiej linii wyświetlacza
\$0C	kod znaku CLS - czyści wyświetlacz, ustawia kursor w lewym górnym rogu
\$0D	kod znaku CR znaczenie jak kodu LF
\$10...\$FD	kody ASCII sterownika HD44780 zawarte w ROM
\$FF	kod danych, następny bajt zostanie zapisany do pamięci danych sterownika LCD, umożliwia to tworzenie własnych znaków, np. sekwencja kodów: FE 40 FF 00 FF 04 FF 08 FF 1F FF 08 FF 04 FF 00 FF 00 zmieni wygląd pierwszego znaku z CGRAM (litera „ą”) na strzałkę skierowaną w lewo.
\$FE	kod sterujący, następny bajt będzie zinterpretowany przez sterownik LCD jako rozkaz (zestawienie rozkazów znajduje się w tab. 2).



Rys. 2. Rozmieszczenie elementów na płytce drukowanej.

zostaną do bufora, a gdy tylko wyświetlacz skończy wykonywać poprzednią operację, dane zostaną do niego wysłane.

Standardowa konfiguracja sterownika

Po włączeniu zasilania, po wykonaniu rozkazu \$FE, \$FE lub \$FE, \$FF sterownik zostanie ustawiony następująco:

- Function Set=%00111000 (dwie linie, matryca znaku 5*7 punktów).
- Entry Mode Set=00000110 (kursor przesuwany w prawo, skrolowanie danych wyświetlacza wyłączone).
- Display ON/OFF control = %00001111 (włączony wyświetlacz, widoczny kursor, miganie znaku w pozycji kursora).
- Do pamięci CG_RAM zostaną skopiowane polskie znaki.

Dzięki temu, że mamy dostęp do wszystkich rozkazów sterownika HD44780, z wyświetlaczem możemy zrobić wszystko to, co umożliwia sterownik. W układach serii EDE nie ma dostępu do pamięci danych, co wyklucza możliwość tworzenia własnych znaków.

Sławomir Skrzyński
slawomir.skrzynski@ep.com.pl

Wzory płytek drukowanych w formie PDF są dostępne w Internecie pod adresem: <http://www.ep.com.pl/?pdf/styczen02.htm> oraz na płycie CD-EP01/2002B w katalogu PCB.

Tab. 2. Kody rozkazów.

Wartość binarna	Opis	Funkcja	
%0 0 0 0 0 0 0 1	Clear display	kasuje wyświetlacz i ustawia kursor pod adres 0	
%0 0 0 0 0 0 1 x	Return home	ustawia kursor pod adres 0	
%0 0 0 0 0 1 i/d s	Entry mode set	ustawia kierunek przesunięcia kursora oraz włącza lub wyłącza przesuwanie danych wyświetlacza: - i/d = 1 (increment) zwiększaj (+1) adres kursora po każdym wpisie lub odczycie danych, - i/d = 0 (decrement) zmniejszaj (-1) adres kursora po każdym wpisie lub odczycie danych, - s=1 przesunięcie wyświetlanych znaków na wyświetlaczu. Poniżej pokazano przykładowe efekty działania różnych ustawień. Na wyświetlaczu znajduje się napis „Ala”, następnie kursor jest ustawiany na pozycję 8 i wpisywany jest napis „Kot”.	
Entry mode set =\$04	Entry mode set =\$05	Entry mode set =\$06	Entry mode set =\$07
%0 0 0 0 1 d c b	Display ON/OFF control	1=on (włącz), 0= OFF (wyłącz): d - cały wyświetlacz c - kursor b - miganie znaku w pozycji kursora	
%0 0 0 1 s/c r/l x x	Cursor and display shift	s/c=1 - obrót danych wyświetlacza s/c=0 - przesunięcie kursora r/l=1 - obrót/przesunięcie w prawo r/l=0 - obrót/przesunięcie w lewo	
Cursor and display shift=\$10	Cursor and display shift=\$14	Cursor and display shift=\$18	Cursor and display shift=\$1C
<p>W rzeczywistości dane w pamięci nie są przesuwane, zmienia się tylko adres tzw. okna wyświetlania (jak na rysunku obok).</p>			
%0 0 1 dl n f x x	Function set	dl=1 - 8-bitowa szyna sterująca dl=0 - 4-bitowa szyna sterująca n=1 - 2 linie wyświetlacza n=0 - 1 linia wyświetlacza f=1 - znaki 5x10 punktów f=0 - znaki 5x7 punktów	
%0 1 a5 a4 a3 a2 a1 a0	Set CG_RAM address	Ustawienie adresu generatora znaków w liczniku adresów. Po tej operacji można pisać lub czytać generator znaków z pamięci RAM	
%1 a6 a5 a4 a3 a2 a1 a0	Set DD_RAM address	Ustawienie adresu danych w liczniku adresów (ustawienie kursora na wybraną pozycję). Sekwencja znaków: FE 82 30 31 FE 8A 32 33 da następujący efekt na wyświetlaczu: 	
%1 1 1 1 1 1 0 1		Ustawia wyjście OUT w stan wysoki.	
%1 1 1 1 1 1 0 0		Ustawia wyjście OUT w stan niski.	
%1 1 1 1 1 1 1 0	Interface reset	Inicjalizacja sterownika. Wysłanie sekwencji rozkazów: FE, FE spowoduje zerowanie sterownika, ustawi się standardowa konfiguracja, bufor RS zostanie wyczyszczony, pojawi się komunikat powitalny.	
%1 1 1 1 1 1 1 1		Inicjalizacja sterownika LCD (sekwencja FE FF).	