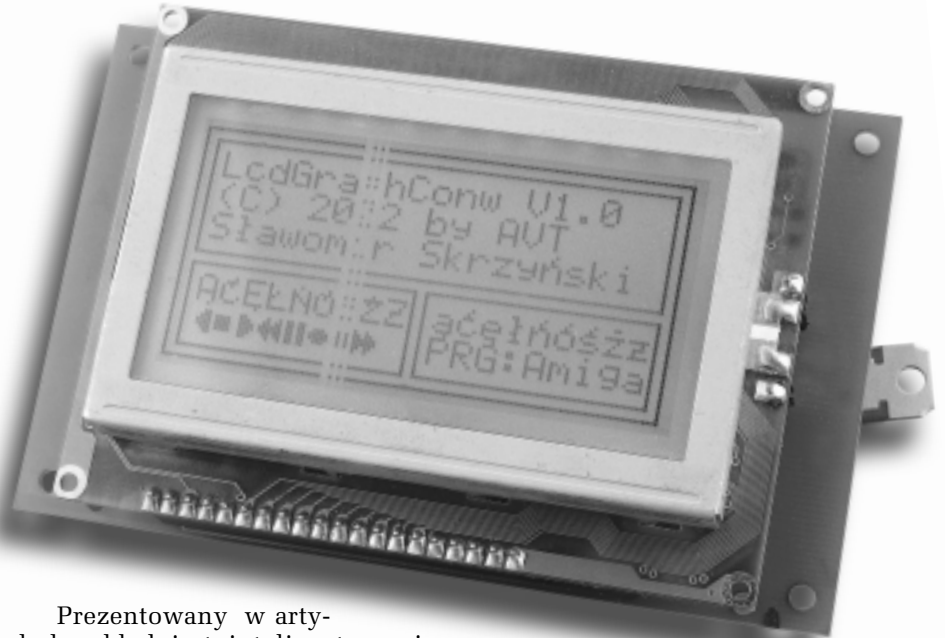


Interfejs graficznych wyświetlaczy LCD

AVT-5093

Sterowanie graficznym wyświetlaczem LCD jest dosyć kłopotliwe, szczególnie gdy wyświetlacz nie ma wbudowanego sterownika z funkcjami ułatwiającymi wyświetlanie oraz rysowanie grafiki i tekstów. W artykule opisano interfejs sprzętowy między wyświetlaczem a sterującym mikrokontrolerem. Dzięki niemu można łatwiej wykonać większość typowych zadań graficznych.

Rekomendacje: artykuł polecamy projektantom zamierzającym stosować wyświetlacze graficzne w swoich opracowaniach.



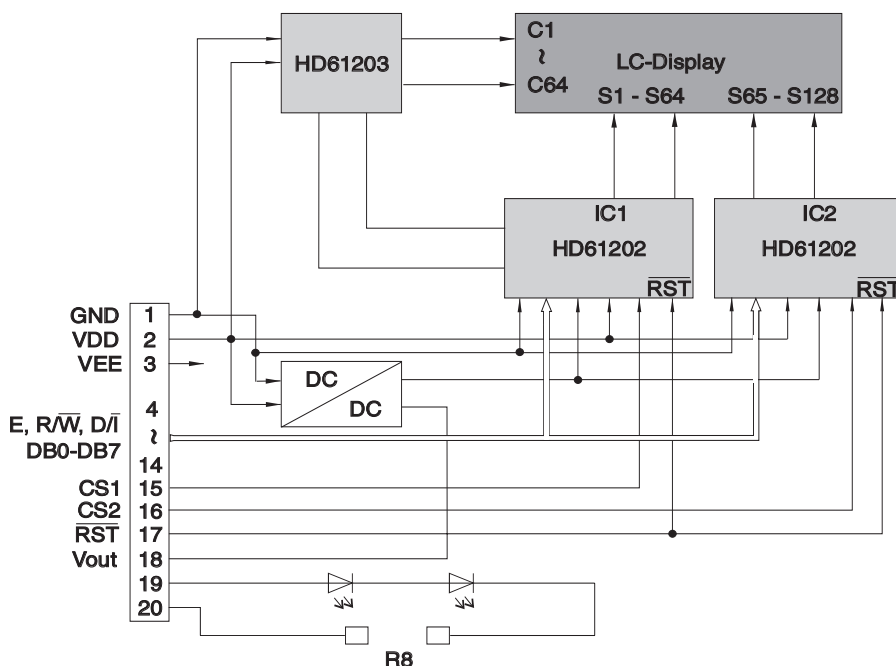
Prezentowany w artykule układ jest inteligentnym interfejsem między sterującym mikrokontrolerem a wyświetlaczem. Dzięki temu projektant może skupić się na przygotowaniu oprogramowania (np. oscyloskopu czy własnego GameBoya) i nie rozpraszać się problemami związanymi z obsługą LCD, a jest ich wiele. Zastosowanie interfejsu

eliminuje wiele problemów związanych z obsługą wyświetlacza, a ponadto, do jego sterowania wystarczy zaledwie jedno wyprowadzenie mikrokontrolera. Zastosowanie transmisji szeregowej nie jest krytyczne dla wyświetlania danych na wyświetlaczach LCD, gdyż są one stosunkowo wolne. Poza tym, odświeżanie obrazu przeprowadza się zwykle wtedy, gdy trzeba go zmienić, ale nie częściej niż kilka razy na sekundę.

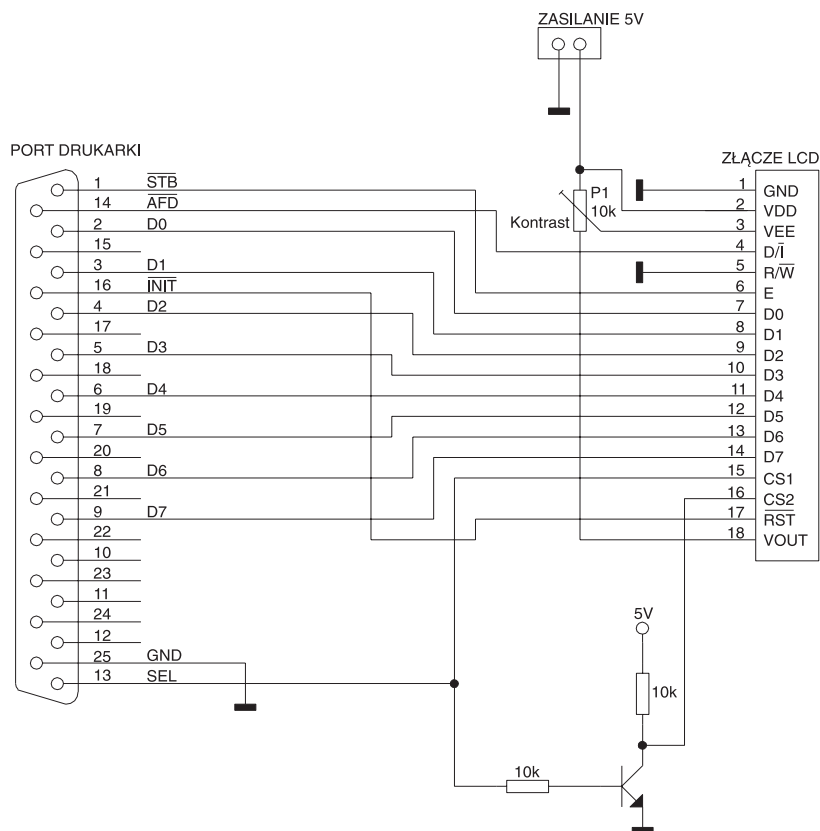
Moduł wyświetlacza

Wyświetlacz zastosowany w projekcie ma oznaczenie EAP128-6N2LED (polecam go ze względu na niską cenę). Matryca ekranu składa się ze 128 x 64 punktów. Ma wbudowany podświetlacz z diod LED i przetwornicę dostarczającą ujemne napięcie polaryzujące. Sterownik wyświetlacza posiada magistralę (tak jak i w LCD alfanumerycznych) zgodną z procesorami Motorola (6809, 680x0, 683xx). Zewnętrzny procesor sterujący ma poprzez tę magistralę dostęp do pamięci ekranu wyświetlacza.

Ekran zawiera 128 x 64 punktów, zgrupowanych w dwóch segmentach po 64 x 64 punkty.



Rys. 1. Schemat blokowy sterownika zastosowanego w wyświetlaczu graficznym opisanym w artykule



Rys. 2. Jeden z możliwych sposobów podłączenia wyświetlacza graficznego do portu drukarkowego PC

Sterowanie wyświetlaniem jest więc podobne jak w wyświetlaczach alfanumerycznych 4x40 znaków, w których zastosowano dwa sterowniki 2 x 40 znaków, programowane przez jedną magistralę. W prezentowanym wyświetlaczu matrycę obsługują także dwa niezależne sterowniki 64 x 64 pun-

ktę. To dodatkowo komplikuje sterowanie wyświetlaczem.

Na rys. 2 przedstawiono schemat układu umożliwiającego dołączenie wyświetlacza do portu drukarkowego komputera PC. Interfejs jest tak prosty, że można go wykonać na płytce uniwersalnej.

Tab. 1. Zestawienie poleceń sterownika HD61208

| Polecenie | Kod | | | | | | | | | | Opis |
|---------------------|-----|----|-----------------------------------|-----|----------------------------------------|-----------------------|-----|-----------------------|--------------------------------------------------|-------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | R/W | DI | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 | |
| Display ON/OFF | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1/0 | Włącza/wyłącza wyświetlanie: 1 -> włączone, 0 -> wyłączone |
| Display start line | 0 | 0 | 1 | 1 | Początkowa linia wyświetlania (0...63) | | | | | | Numer linii wyświetlanej w górnej części ekranu |
| Set page(X address) | 0 | 0 | 1 | 0 | 1 | 1 | 1 | Współrzędna X (0...7) | | | Ustala numer aktywnej strony pamięci obrazu (współrzędna X) |
| Set address | 0 | 0 | 0 | 1 | Współrzędna Y (0...63) | | | | | | Ustala początkowy adres w liczniku wskazującego współrzędną Y |
| Status Read | 1 | 0 | B U S Y | 0 | ON / OFF | R E S E T | 0 | 0 | 0 | 0 | Odczyt rejestru statusu RESET: 1 -> zerowanie, 0 -> praca ON/OFF: 1 -> wyświetl., 0 -> wyświetl. wyłączony BUSY: 1 -> kontroler zajęty, 0 -> kontroler gotowy |
| Write display data | 0 | 1 | Dane wpisywane do pamięci obrazu | | | | | | Zapis 8-bitowego słowa danych do pamięci obrazu | Dostęp (zapis/odczyt) do pamięci obrazu. Po każdej operacji adres Y jest automatycznie inkrementowany | |
| Read display data | 1 | 1 | Dane odczytywane z pamięci obrazu | | | | | | Odczyt 8-bitowego słowa danych do pamięci obrazu | | |

W przypadku dołączenia wyświetlacza do magistrali procesora innego niż Motorola należy pamiętać, że sterownik wyświetlacza wymaga, aby sygnały RW, DI i CS ustaliły się przed wystąpieniem poziomu aktywnego sygnału E. Aby spełnić ten warunek, najlepiej sygnały sterujące dla wyświetlacza wytworzyć w układzie według rys. 3. Powoduje to, że obsługa wyświetlacza jest trochę nienaturalna (inne adresy do zapisu, inne do odczytu), ale mamy gwarancję, że sygnał RW zostanie ustalony przed pojawieniem się sygnału E.

Sterowanie wyświetlaczem

W tab. 1 opisano sposób dostępu do rejestrów kontrolnych i pamięci. Wybór operacji odbywa się poprzez ustawienie odpowiednich stanów logicznych na wejściach sterujących kontrolera wyświetlacza:

- odczyt rejestru kontrolnego: (DI=L, RW=H) - znaczenie poszczególnych bitów jest następujące:
bit 7 - BUSY

BUSY=L - sterownik gotowy do przyjęcia rozkazu

BUSY=H - sterownik realizuje rozkaz, CPU musi czekać

bit 6 - zawsze L

bit 5 - ON/OFF

ON/OFF=L - wyświetlacz wyłączony

ON/OFF=H - wyświetlacz włączony

bit 4 - RESET

RESET=L - normalna praca

RESET=H - wyświetlacz w stanie zerowania (aktywny stan linii zerującej)

bit 3...0 - zawsze L

- zapis rejestru kontrolnego: (DI=L, RW=L) - znaczenie poszczególnych bitów jest następujące:
0011111x - włączenie/wyłączenie wyświetlacza

x=H - włączenie wyświetlacza

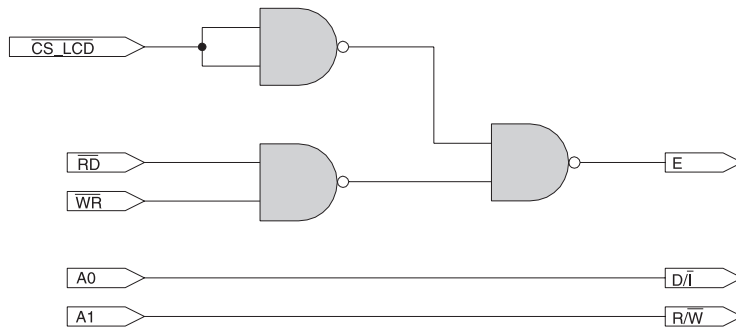
x=L - wyłączenie wyświetlacza

01yyyyyy - ustawienie numeru bajtu na stronie (współrzędna pozioma) w przedziale 0...63

10111xxx - ustawienie numeru strony w przedziale 0...7

11yyyyyy - ustawienie offsetu w przedziale 0...63 - polecenie to umożliwia przewijanie ekranu

odczytuje bajt z pamięci ekranu - odczytuje bajt ustawiony rozkazami 01yyyyyy i 10111xxx,



UWAGA: w wyświetlaczach alfanumerycznych sygnał D/I nosi nazwę RS

Rys. 3. Interfejs ułatwiający dołączenie wyświetlacza graficznego do mikrokontrolera

zapis bajtu do pamięci ekranu
 - zapisuje bajt ustawiony rozkazami 01yyyyyy i 10111xxx.

Na rys. 3 przedstawiono organizację pamięci ekranu wyświetlacza. Aby wyświetlić na ekranie punkt o współrzędnych XY, należy postępować następująco: współrzędną Y dzielimy przez 8 i tak uzyskana część całkowita wyniku określa numer strony pamięci ekranu. Reszta z dzielenia jest numerem bitu, który chcemy zmodyfikować. Współrzedną X porównujemy z liczbą 63. Jeśli jest większa, to oznacza, że obsługujemy drugi segment wyświetlacza. Adres punktu i numer bitu mamy obliczony, należy więc odczytać zawartość pamięci ekranu LCD i przeprowadzić operację OR (w przypadku wyświetlania punktu) lub AND (w przypadku wygaszenia) na wyliczonym numerze bitu.

Aby zainicjować wyświetlacz, należy wysłać następujące rozkazy:

```
MovCTR_LCD, #00111111
;włączenie LCD
MovCTR_LCD, #11000000
;zerowy offset
```

Aby wyświetlić punkt w lewym górnym rogu wyświetlacza (w jednej połówce), należy wysłać następujące rozkazy:

```
MovCTR_LCD, #10111000
;ustawienie nr strony
;pamięci ekranu
MovCTR_LCD, #01000000
;ustawienie adresu
;w pamięci ekranu
MovDATA_LCD, #00000001
;zapis do pamięci ekranu
```

Aby wyświetlić punkt o współrzędnych X=10, Y=20, należy wysłać następujące rozkazy:

```
MovCTR_LCD, #10111000+2
;ustawienie nr strony
;pamięci ekranu
```

```
MovCTR_LCD, #01000000+10
;ustawienie adresu
;w pamięci ekranu
MovDATA_LCD, #00010000
;zapis do pamięci ekranu
```

Aby na wyświetlaczu LCD wyświetlić znak, należy jego wzór skopiować, linia po linii, z pamięci zawierającej wzory znaków do pamięci ekranu. Przykładowy program w Pascalu (także jego wersja źródłowa) sterujący wyświetlaczem przez port równoległy jest dostępny na CD-EP12/2002B i na naszej stronie WWW.

Ponieważ nie ma możliwości odczytania danych z kontrolera LCD, program nie weryfikuje stanu linii busy, tylko odczekuje katalogowy czas (właściwie Pascal działa tak wolno, że nie trzeba czekać). W razie potrzeby opóźnienie można wstawić w procedurze STROB. W pamięci komputera znajduje się wirtualny ekran, z którego dane są kopiowane do LCD. Jeśli nie trzeba zapisywać numeru strony lub adresu bajtu do LCD, to procedury te są pomijane. Dzięki temu przyspieszono wykonywanie operacji zapisu.

Przykładowy program wyświetla punkty, rysuje prostokąt, wyświetla znaki alfanumeryczne i rysuje linię pod dowolnym kątem. Stałe FLIP_X, FLIP_Y i ORIENT umożliwiają zmianę orientacji ekranu. Gdy ustawimy FLIP_X i FLIP_Y na 1, ekran zostanie odwrócony „do góry nogami”. Ustawienia na 1 stałej ORIENT umożliwia wyświetlanie danych w pionie, a nie w poziomie (jak w większości automatów do gier).

Aby przybliżyć Czytelnikom obsługę wyświetlacza, na list. 1 przedstawiam najistotniejszy fragment procedury wyświetlania

punktu na LCD napisanej w assemblerze 8051. Plik z kodem źródłowym jest dostępny na stronie EP i na płycie CD-EP12/2002B.

Charakterystyka interfejsu

Program sterujący pracą mikrokontrolera realizuje następujące funkcje:

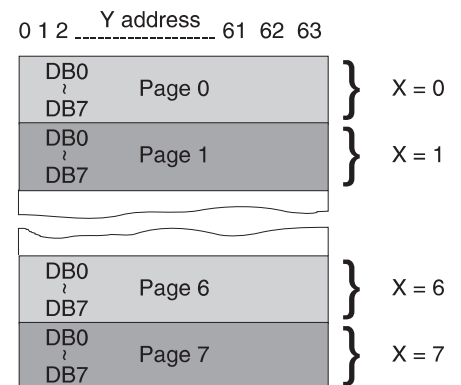
- po wyzerowaniu wyświetla logo do czasu otrzymania dowolnego znaku przez RS232,
- steruje podświetlaniem LCD,
- wyświetla punkty,
- wyświetla pojedyncze znaki (także polskie),
- wyświetla teksty,
- przewija ekran w pionie,
- czyści ekran,
- przełącza wyświetlany obraz w negatyw,
- zmienia orientację ekranu w osi X,
- zmienia orientację ekranu w osi Y,
- obraca ekran o 90 stopni.

Nie została zaimplementowana procedura rysowania linii, ponieważ wymaga ona wykonywania operacji zmiennoprzecinkowych. Są natomiast dostępne wzory wszystkich polskich znaków, ramek i znaki używane w odtwarzaczach CD.

Interfejs akceptuje sygnały w standardzie TTL i RS232C. Można go wyzerować, wymuszając poziom niski na wyprowadzeniu 4 złącza J3.

Budowa i działanie interfejsu

Schemat elektryczny interfejsu przedstawiono na rys. 5. Napięcie zasilające wyświetlacz jest podawane na złącze szpilkowe J3. Na to złącze wprowadzane są dane z układu sterującego i sygnał ze-



Rys. 4. Mapa pamięci sterownika wyświetlacza graficznego ze sterownikiem HD61208

List. 1.

```

;Wejscie:
;R2 - współrzędna X
;R3 - współrzędna Y
;R4 - 1-zapala punkt, 0-gasi punkt
;Zmienia: -
;
;
; lcd_point: mova,R2 ;Która polowke obsługujemy?
; cjne a,#(MAX_X/2)+1,$+3 ; czy przejście na druga polowke?
; jnclcd_point_cs0
; clr_CS ; - ustawienie pierwszej polowki
; sjmp lcd_point_cs1
; lcd_point_cs0: setb _CS ; - tak wiec ustawienie drugiej polowki
; lcd_point_cs1:
; mova,R3 ;Obliczenie adresu strony
; movB,#8 ; Y/8
; divab ; reszta z dzielenia w B
; call lcd_page
; mova,R2 ; X
; call lcd_adr_y
; mova,B ; reszta z dzielenia do Acc
; cjne R4,#0,lcd_point_set1 ;Jesli SET punktu to skok
; movdptr,#tab_and ; - nie set wiec RESET
; sjmp lcd_point_set2
; lcd_point_set1: movdptr,#tab_or
; lcd_point_set2: movc a,@a+dptr ;Wczytanie danych z tablicy
; movB,a ; nr bitu do zapalenia zapamiętaj w B
; call rd_dana ;Odczytaj zawartosc pamieci ekanru w miejscu zmiany
; call rd_dana ; korekta (odczytuje Y-1)
; push ACC ; zapamiętaj odczytana wartosc z pamieci ekranu
; mova,R2 ; X
; call lcd_adr_y
; popACC ;Odtworzenie odczytanej danej z LCD
; cjne R4,#0,lcd_point_set3 ;Jesli SET punktu to skok
; anla,B
; sjmp lcd_point_set4
; lcd_point_set3: orla,B
; lcd_point_set4: call wr_dana
;
;
; lcd_point_exit: ret
;
;
; tab_or: DB 01h,02h,04h,08h,10h,20h,40h,80h
; tab_and: DB 11111101b
; DB 11111101b
; DB 11111011b
; DB 11110111b
; DB 11101111b
; DB 11011111b
; DB 10111111b
; DB 01111111b
    
```

rujący. Dane przesyłane szeregowo są podawane na wejście bramki U2C układu 74HC14. Rezystor R1 zabezpiecza wejście bramki w przypadku wystąpienia na nim napięcia większego od napięcia zasilania lub napięcia ujemnego. Dzięki temu można doprowadzić do wejścia układu napięcia zgodne ze standardem RS232C lub o poziomach TTL. Jeśli sygnał ma poziomy TTL, jumper na J2 musi zwierać styki 2-3, a jeśli sygnały są zgodne z RS232C, na J2 zwieramy styki 1-2. Dane są podawane na wejście RxD procesora. Elementy C1 i R6 tworzą obwód zerujący procesor po włączeniu zasilania. Mikrokontroler można

wyzerować, wymuszając poziom niski na styku 4 złącza J3.

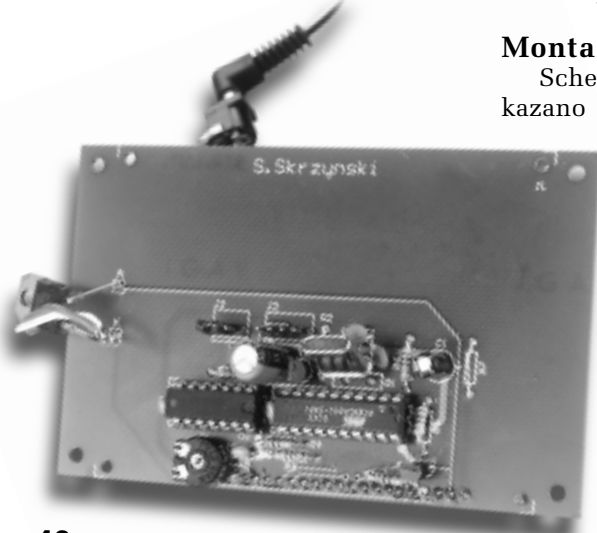
Mikrokontroler steruje wyświetlaczem LCD za pośrednictwem większości wyprowadzeń portów P1 i P3. Rezystory R3 i R4 są konieczne, ponieważ wyprowadzenia P1.0 i P1.0 mikrokontrolera nie zawierają wewnętrznych obwodów podciągania. Tranzystor T2 załącza podświetlenie wyświetlacza. Niski poziom na wyprowadzeniu 3 procesora powoduje włączenie tego tranzystora. Za pomocą bramki U2A steruje się wejściem CS1 kontrolera LCD w przeciwfazie z wejściem CS2. Potencjometrem P1 reguluje się kontrast wyświetlacza.

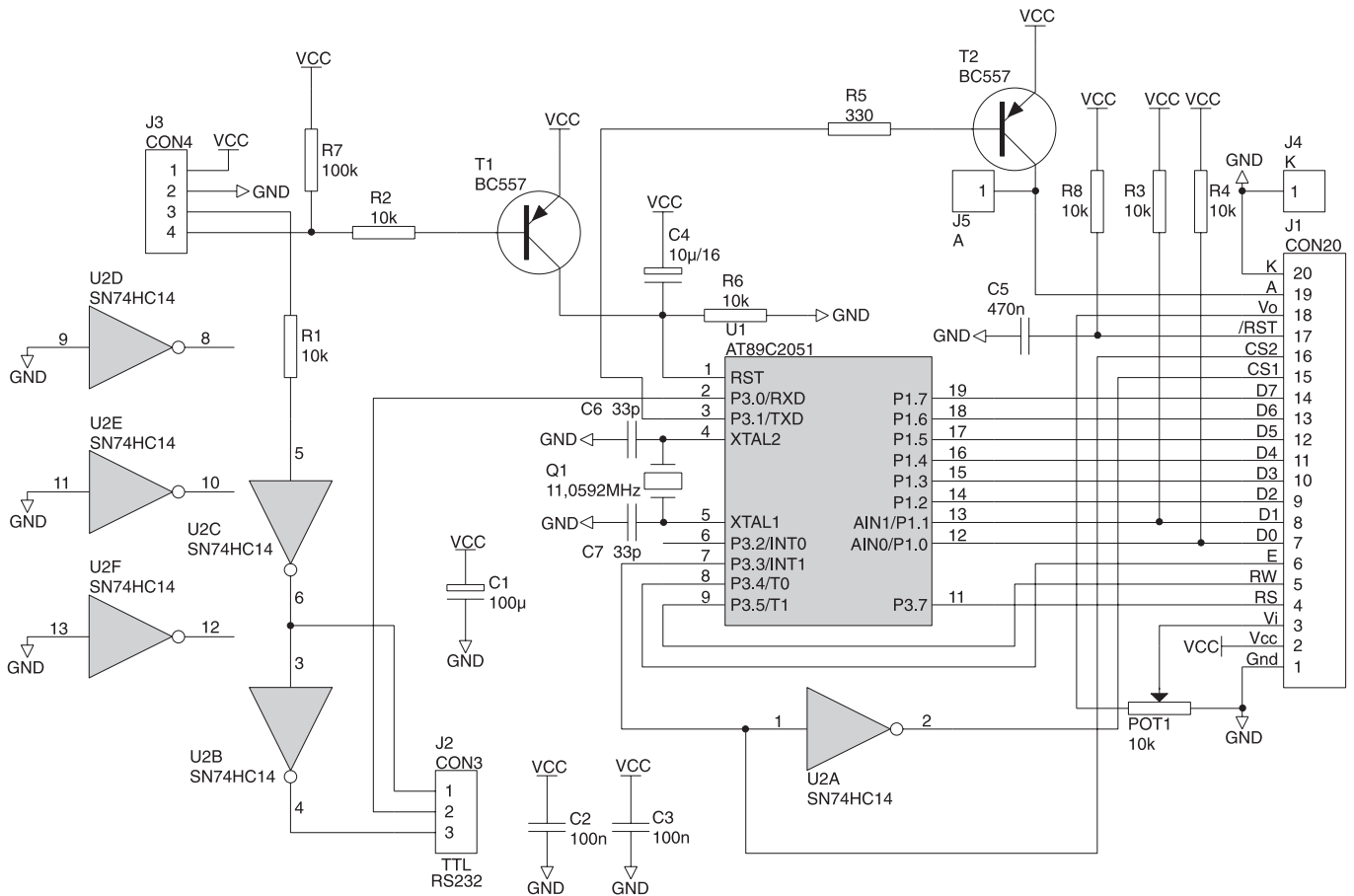
Montaż i uruchomienie

Schemat montażowy płytki pokazano na rys. 6. W pierwszej kolejności montujemy rezystory, podstawki pod układy scalone, kondensatory i potencjometr. Następnie lutujemy złącza: J2 - listwa goldpin 1x3, J3 - listwa goldpin 1x4, J1 - listwa goldpin 20x1 (montujemy je od strony druku). Podobnie postępujemy z gniazdami tulipa-

nowymi J4 i J5. Umieszczamy układ U2 w podstawce i włączamy zasilanie. Na styk 3 złącza J3 podajemy szeregowo dane z PC - na wyprowadzeniu 2 układu U1 powinny występować impulsy. Po zwarceniu styku 4 złącza J3 z masą, na wyprowadzeniu 1 podstawki pod U1 powinien być poziom „H“. Po tych czynnościach można umieścić mikrokontroler i wyświetlacz w miejscach dla nich przeznaczonych. Po włączeniu zasilania na wyświetlaczu powinno pojawić się logo i włączyć podświetlenie wyświetlacza. Logo zniknie i podświetlenie wyłączy się po odebraniu pierwszego znaku. Jeśli logo jest mało widoczne, regulujemy kontrast wyświetlacza. Jeśli regulacja nie jest skuteczna, sprawdzamy, czy na styku 18 złącza J1 występuje napięcie ujemne (około -9V). Brak napięcia świadczy o uszkodzeniu wyświetlacza lub zwarceniu ścieżek prowadzących do potencjometru. Jeśli napięcie ma prawidłową wartość, prawdopodobnie nie pracuje procesor. W pierwszej kolejności sprawdzamy, czy na wyprowadzeniu 1 mikrokontrolera jest poziom „L“. Jeśli występuje poziom „H“, uszkodzony może być tranzystor T1 lub rezystor R6. Jeśli jest poziom „L“, zwieramy na chwilę styk 4 złącza J3 z masą. Na wyprowadzeniu 1 mikrokontrolera powinien pojawić się na chwilę poziom „H“. Jeśli po ponownym zerowaniu nie pojawi się logo, to nie pracuje poprawnie mikrokontroler. Sprawdzamy, czy na wyprowadzeniu 4 mikrokontrolera występuje fala prostokątna. Podczas testu należy używać sondy 1:10, aby nie przeciążyć oscylatora, co spowodowałoby zerwanie drgań.

Jeśli układ nadal nie pracuje poprawnie, należy sprawdzić, czy po zerowaniu pojawiają się impulsy lub poziom niski na wyprowadzeniu 8 mikrokontrolera. Jeśli są impulsy, to sprawdzamy, czy występują również na wyprowadzeniach 1 i 2 układu U2. Należy pamiętać, że impulsy pojawiają się tam na krótko po zerowaniu. Jeżeli występują impulsy na wyprowadzeniach U2, a nie jest wyświetlane logo na LCD, świadczy to o uszkodzeniu wyświetlacza.





Rys. 5. Schemat elektryczny interfejsu wyświetlacza LCD

Można jeszcze sprawdzić, czy wyprowadzenia mikrokontrolera 11 i 13 są poprawnie „podciągnięte“ do +5V. Gdy po zerowaniu wyprowadzenie 8 mikrokontrolera na chwilę przyjmuje poziom niski, a później utrzymuje się na poziomie wysokim, oznacza to, że wyświetlacz znajduje się w stanie WAIT. Prawdopodobną przyczyną jest uszkodzenie sterownika LCD.

Przed uznaniem LCD za niesprawny należy jeszcze sprawdzić, czy na styku 17 złącza J1 występuje poziom „H“ - poziom „L“ może blokować pracę LCD.

Programowa obsługa interfejsu

Do pierwszych testów proponuję podłączenie sterownika do komputera z uruchomionym programem terminalowym. Zworka na J2 musi zwierać styki 1-2. W programie terminala ustawiamy następujące parametry transmisji: 4800/8n1, sterowanie przepływem - sprzęt.

Aby wyświetlić punkt, należy wysłać przez RS komendę:
p XX YY,
gdzie:

XX, YY - odpowiednio współrzędna X, współrzędna Y (szesnastkowo po dwa znaki ASCII) - np.: sekwencja p0312 spowoduje wyświetlenie punktu na ekranie o współrzędnych x=3, y=12.

Przekroczenie zakresu współrzędnych (wyjście poza ekran) niczym nie grozi, po prostu rozkaz nie zostanie wykonany. Procedura wyświetlania punktu jest podstawą dla wszystkich innych operacji. Każdą operację można przerwać, wysyłając znak ESC. Ze względu na specyfikę wyświetlacza i programu nie należy umieszczać punktów o współrzędnej Y=0.

Aby wyświetlić znak alfanumeryczny, należy wysłać komendę:
c XX YY CHAR,
gdzie:

XX i YY - współrzędne lewego górnego rogu znaku
CHAR - znak - np.: c0001a spowoduje wyświetlenie litery „a“ w lewym górnym rogu ekranu.

Aby wyświetlić tekst, należy wysłać komendę:

t XX YY CHAR CHAR CHAR...
CHAR CR+LF,
gdzie:

XX i YY to współrzędne lewego górnego rogu tekstu,
CHAR - znaki tekstu,
CR+LF - znak końca linii.

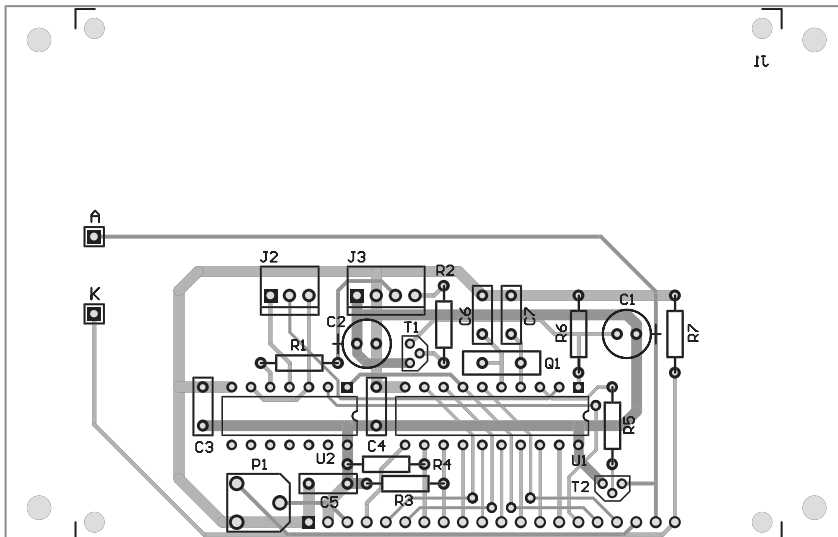
Np.: t0001Ala ma kota [Enter] umieści w lewym górnym rogu tekst „Ala ma kota“. Znakiem końca linii może być kod LF (\$0A), CR (\$0D) lub CR+LF (\$0D0A). Wpisanie tekstu można też zakończyć, wysyłając kod ESC (\$1B).

Do przewijania ekranu w pionie służy komenda:

s YY,
gdzie YY - oznacza przesunięcie ekranu w stosunku do punktu 00.

Wysyłając w równych odstępach czasu rozkaz „s YY“, gdzie YY będzie zmieniać się od 0 do 63, uzyskamy płynne przewijanie całego ekranu z góry na dół. Niestety wyświetlacz jest dość wolny i ciągłe przewijanie nie wygląda zbyt dobrze.

Ekran czyścimy, wysyłając literę C. Ekran pozytywowo (ustawienie domyślne) uzyskujemy, wysyłając z01, negatywowo z00. Obrócenie obrazu ekranu w pozio-



Rys. 6. Rozmieszczenie elementów na płycie drukowanej interfejsu

mie uzyskujemy komendą x01, ekran normalny x00. Obrócenie obrazu ekranu w pionie uzyskujemy komendą y01, ekran normalny y00. Zmianę orientacji ekranu uzyskamy komendą o01, orientacja normalna o00.

Zmiana orientacji ekranu lub obrócenie obrazu ekranu w osi X czy Y dotyczy znaków, które wpiszemy po wysłaniu odpowiedniej komendy. Inaczej mówiąc, komendy te zmieniają sposób obliczania współrzędnych podczas wyświetlania punktu. Włączenie podświetlenia LCD uzyskamy po wydaniu komendy: L01, zgaszenie: L00.

Polskie znaki diaktryczne oraz znaki specjalne mają następujące kody (szesnastkowo): Ā - 01, Ć - 02, Ę - 03, Ł - 04, Ń - 05, Ó - 06, Ś - 07, Ź - 08, Ż - 09, Ą - 11, Ć - 12, Ę - 13, Ł - 14, Ń - 15, Ó - 16, Ś - 17, Ź - 18, Ż - \$19, zanegowana

spacja - 80, odtwarzanie 81, odtwarzanie do tyłu 82, szybko naprzód 83, szybko w tył 84, stop 85, pauza 86, zapis 87, ramki 88...92.

Udostępniamy wszystkie programy w postaci źródłowej, dzięki czemu można je dołączyć do swojego oprogramowania. Aby ułatwić wykorzystanie procedur, poniżej znajdują się ich opisy:

```
lcd_clr:
;Czysci cały LCD
;WE: R4 - wzor wypelnienia
;Zmienia: Acc, R7, R6
```

```
lcd_point
;Zapala/gasi punkt
;R2 - X
;R3 - Y
;R4 - 1-zapala, 0-gasi
;Zmienia: -
```

```
lcd_char
;Wypisuje znak na wyswietlacz
```

WYKAZ ELEMENTÓW

Rezystory

R1...R4, R6, R8: 10kΩ
R5: 330Ω
R7: 100kΩ
Pot1: potencjometr 10kΩ

Kondensatory

C1: 100μF
C2, C3: 100nF
C4: 10μF/16V
C6, C7: 33pF
C8: 470nF

Półprzewodniki

T1, T2: BC557
U1: AT89C2051
U2: 74HC14

Różne

J1: listwa goldpin 20x1
J2: listwa goldpin 3x1
J3: listwa goldpin 4x1
Q1: 11.0592MHz
Wyświetlacz graficzny P128-6n2

```
;Acc- kod znaku
;R2 - X
;R3 - Y
;R4 - 1-zapala, 0-gasi
```

```
lcd_text
;Wypisuje tekst na wyswietlacz
;R2 - X
;R3 - Y
;R4 - 1-zapala, 0-gasi
;DPTR - adres tekstu
;          zakonczony 0
```

Sławomir Skrzyński, AVT
slawomir.skrzynski@ep.com.pl

Wzory płytek drukowanych w formacie PDF są dostępne w Internecie pod adresem: <http://www.ep.com.pl/?pdf/grudzien02.htm> oraz na płycie CD-EP12/2002B w katalogu PCB.