

*Pamiętam pierwsze wrażenie, jakie wywarł na mnie wyświetlacz VFD. Zachwycił mnie przede wszystkim doskonale czytelny z odległości nawet kilku metrów obraz. Oczywiście wyobraźni już widziałem go w projektowanych przeze mnie sterownikach urządzeń.*



## Sterowanie alfanumerycznych wyświetlaczy VFD

### Wyświetlacze VFD

Skrót VFD pochodzi od angielskich wyrazów *Vacuum Fluorescent Display*. Wyświetlacz VFD to rodzaj trójelektrodowej lampy próżniowej,

w której poszczególnymi elektrodami są:

- katoda: cienkie druty (żarzone) znajdujące się nad świecącymi obszarami,

- siatka kontrolna umieszczona pomiędzy katodą a matrycą znaku (kontrolująca świecenie lub nie punktów czy segmentów),
- anoda: świecąca warstwa tzw. lu-

**Tab. 1. Wykaz rozkazów akceptowanych przez wyświetlacz CU20025-U2J firmy Noritake - Itron**

Instrukcja	Kod instrukcji										Opis
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Clear display (kasowanie ekranu)	0	0	0	0	0	0	0	0	0	1	Kasowanie ekranu oraz zapis 0 do wskaźnika adresu DD RAM (pamięć znaków)
Cursor home (powrót kursora do wsp. 0,0)	0	0	0	0	0	0	0	0	1	x	Ustawienie wskaźnika adresu DD RAM na wartość 0 i powrót kursora do pozycji początkowej. Powoduje również przywrócenie stanu przesuwanego obrazu. Zawartość DD RAM pozostaje niezmienną
Entry mode set (ustawienie trybu dla znaków)	0	0	0	0	0	0	0	1	I/D	S	Ustawia kierunek ruchu kursora oraz sposób przesuwania obrazu w czasie zapisu/odczytu danych I/D: 1 to automatyczne zwiększanie, 0 - zmniejszanie adresu S: 1 to przesuwanie ekranu dozwolone, 0 - zabronione
Display ON/OFF controll (kontrola wyświetlania)	0	0	0	0	0	0	1	D	C	B	Załączenie/wyłączenie obrazu, kursora oraz migotania kursora na pozycji znaku D: 1 - ekran załączony, 0 - ekran wyłączony C: 1 - kursor załączony, 0 - kursor wyłączony B: 1 - migotanie załączone, 0 - migotanie wyłączone
Cursor or display shift (przesuwanie kursora/obrazu)	0	0	0	0	0	1	S/C	R/L	x	x	Przesuwa kursor lub zawartość ekranu nie zmieniając zawartości DD RAM S/C: 1 - przesuwanie obrazu, 0 - przesuwanie kursora R/L: 1 - kierunek przesunięcia w prawo, 0 - w lewo
Function set (słowo 4-/8-bitowe)	0	0	0	0	1	IF	x	x	x	x	Ustawia długość słowa danych dla interfejsu IF: 1 - interfejs 8 bitów, 0 - interfejs 4 bity
Brightness controll (jasność świecenia)	1	0	x	x	x	x	x	x	BR1	BR0	Polecenie akceptowane po rozkazie "Function Set" jako bajt kontroli jasności świecenia. BR1, BR0 = 00: 100%, 01: 75%, 10: 50%, 11: 25%
CG RAM address setting (nastawy adresu CG RAM)	0	0	0	1	Adres CG RAM					Ustawia adres CG RAM (pamięci generatora znaków, np. przy definiowaniu własnych znaków)	
DD RAM address setting (nastawy adresu DD RAM)	0	0	1	Adres DD RAM					Ustawia adres pamięci DD RAM (pamięci obrazu, np. przy realizacji funkcji umieszczającej znak na współrzędnych [kolumna, wiersz])		
Busy flag and address reading (odczyt flagi zajętości i adresu)	0	1	BF	Wartość wskaźnika adresu					Odczyt flagi zajętości (gdy BF = 1, to realizowane jest przetwarzanie wewnętrzne i kontroler nie przyjmuje danych) oraz wskaźnika adresu DD RAM		
Data writing to CG or DD RAM (zapis bajtu do CG RAM lub DD RAM)	1	0	Dane zapisywane					Zapis danych do DD RAM lub CG RAM w zależności od tego, czy ostatnio wykonywano polecenie Set DD RAM Address, czy Set CG RAM Address			
Data reading from CG or DD RAM (odczyt bajtu z CG RAM lub DD RAM)	1	1	Dane odczytywane					Odczyt danych z DD RAM lub CG RAM w zależności od tego, czy ostatnio wykonywano polecenie Set DD RAM Address, czy Set CG RAM Address			

List. 1. Fragment programu w języku C opisywanego w EP 7-8/2002 przeznaczonego do sterowania wyświetlaczem z kontrolerem HD44780 zawierający najważniejsze funkcje

```
// zapis bajtu do lcd
void WriteByteToLcd(char X)
{
    P2 |= 0xF0;           //ustawienie górnej połówki portu P2 na "1"
    P2 &= (X | 0x0F);     //„bezkolizyjny” zapis 1-szej połówki bajtu
                          //(przez funkcję logiczną)
    LcdEnable = 0;       //zapis do wyświetlacza (opadające zbocze sygnału E)
    LcdEnable = 1;       //zapis 2-giej połówki bajtu
    X <<= 4;             //przesunięcie 4x w lewo
    P2 |= 0xF0;           //ustawienie górnej połówki portu P2 na "1"
    P2 &= (X | 0x0F);     //zapis 2-giej połówki bajtu
    LcdEnable = 0;       //opadające zbocze E - zapis do LCD
    Delay(1);
}

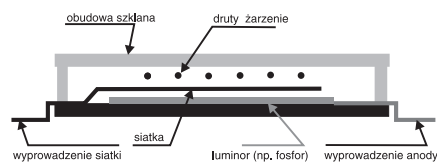
// zapis bajtu do rejestru kontrolnego LCD
void WriteToLcdCtrlRegister(char X)
{
    LcdReg = 0;          //ustawienie sygnałów sterujących
    LcdRead = 0;
    LcdEnable = 1;
    WriteByteToLcd(X);
}

// zapis bajtu do wyświetlacza
void LcdWrite(char X)
{
    LcdReg = 1;
    LcdRead = 0;
    LcdEnable = 1;
    WriteByteToLcd(X);
}

//inicjalizacja wyświetlacza LCD w trybie 4 bity
void LcdInitialize(void)
{
    char i;

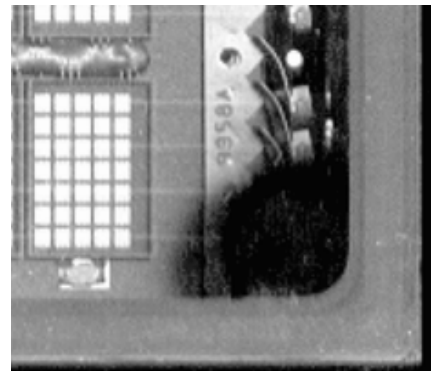
    Delay(15);
    LcdReg = LcdEnable = LcdRead = 0; //wyzerowanie linii LcdReg,LcdRead,LcdEnable
    for (i = 0; i<3; i++)
    {
        LcdEnable = 1;           //impuls na E
        PORT &= 0x3F;           //ustawienie wart. inicjującej
        LcdEnable = 0;
        Delay(5);
    }
    LcdEnable = 1;             //wpisanie wartości 2 do rej. kontr.
    PORT &= 0x2F;             //tylko "górne" 4 bity
    LcdEnable = 0;
    Delay(1);
    WriteToLcdCtrlRegister(0x28); //interfejs 4 bity, znaki 5x7
    WriteToLcdCtrlRegister(0x08); //wyłączenie LCD
    WriteToLcdCtrlRegister(0x01); //kasowanie ekranu, powrót do spoczynkowej
    WriteToLcdCtrlRegister(0x06); //przesuwanie kursora z inkrementacją
    WriteToLcdCtrlRegister(0x0C); //załączenie wyświetlacza
}
.....
```

minoforu (najczęściej jest nim fosfor lub jego związki).  
Budowę wyświetlacza VFD pokazano na rys. 1. Jak w każdej lampie elektronowej, wymagane jest podgrzanie katody, ponieważ wskutek zachodzącej wówczas termoemisji elektronów z katody możliwa jest praca lampy przy niezbyt wysokim napięciu anodowym. W wyświetlaczach



Rys. 1. Budowa wyświetlacza VFD

VFD cienki drut żarnika jest jednocześnie katodą - w lamie tej zastosowano tzw. żarzenie bezpośrednie. Pobierany do rozgrzania katody prąd żarzenia jest przyczyną, że VFD po-



Fot. 2. Widoczny na zdjęciu ciemny „kleks” zmieni swą barwę, jeśli wyświetlacz utraci szczelność i do wnętrza dostanie się powietrze

biera więcej prądu niż równoważny mu funkcjonalnie wyświetlacz LCD. W praktyce wartość ta dla wyświetlacza znakowego wynosi kilkaset mA, a dla graficznego nawet około 1 A.

Elektrony wyemitowane z katody przyciągane są przez anodę, a ich przepływ jest sterowany napięciem siatki. Im mniej ujemny potencjał ma siatka, tym strumień elektronów płynący przez próżnię od katody do anody jest większy.

Luminofor pokrywający anodę świeci bombardowany strumieniem tych elektronów. Gdy na siatce jest potencjał silnie ujemny, elektrony są zawracane w kierunku katody: luminofor nie świeci. Mimo iż opisane działanie segmentu wskaźnika jest bardzo podobne do działania triody, to jednak wyświetlacz różni się od niej sposobem sterowania przepływem prądu anodowego: siatka sterująca działa jak przełącznik, a nie jak regulator. Pewna możliwość wpływu na wartość prądu jest często wykorzystywana przez producentów wyświetlaczy do zmiany jasności świecenia znaków.

Każdy ze znaków uformowany jest z świecących segmentów lub punktów. Typowo, na pojedynczy znak wyświetlacza alfanumerycznego przewidziano matrycę 5 x 7 punktów. Każdy z nich jest miniaturową anodą z doprowadzonym napięciem zasilania.

List. 2. Przykład fragmentu programu sterującego wyświetlaczem LCD lub VFD w języku Bascom

```
'konfiguracja wyświetlacza LCD
Config Lcd = 16 * 1
'wybór sposobu podłączenia
Config Lcdpin = Pin, Db4 = Porta.5, Db5 = Porta.4, Db6 = Porta.3, Db7 = Porta.2,
E = Porta.6,
Rs = Porta.7

'program główny
Do
    Call Gettime
    Locate 1, 1: Lcd Bcd(h); ":"; Bcd(m); ":"; Bcd(s)
Loop
End
```

List. 3. Fragment programu napisanego w języku C do obsługi wyświetlacza VFD

```

/*****
  Obsługa wyświetlacza VFD firmy
  Noritake VFD z użyciem UART
  *****/
Raisonance C module
Uwaga:
Ustaw "Initial Timer 1 value" to 0xFD !!!
(options > project > LX51 > linker >
timer 1 initial value = FD)
Dla rezonatora 11.0592MHz, prędkość
UART wyniesie 9600 bps
*/

#include <reg52.h>
#include <stdio.h>

//inicjalizacja VFD (kasowanie ekranu,
//powrót do pozycji HOME
void VFD_Init(void);
{
    putchar(0x1B);
    putchar(0x49);
}

//ustawienie kursora na pozycji x, y
void GotoXY(char x, char y)
{
    char addr;

    addr == y * 20 + x - 1;
    putchar(0x1B);
    putchar(0x48);
    putchar(addr);
}

//program główny
void main()
{
    VFD_Init();
    printf("%s\n", "Noritake VFD");
    while(1);
}

```

Tak jak w lampie, elektrody wyświetlacza zamknięte są w szklanej bańce, wewnątrz której panuje próżnia. Wyświetlacz, który z jakichś powodów utraci próżnię, łatwo jest rozpoznać: znajdujący się wewnątrz związek chemiczny (tzw. pochłaniacz gazów szkodliwych - *getter*) zmienia swój kolor ze srebrnego (lub ciemnoszarego) na biały, utleniając się pod wpływem powietrza atmosferycznego. W różnych wyświetlaczach srebrny „kleks“ (fot. 2) można znaleźć w różnych miejscach. Niektóre z nich mają gaś obok pola odczytowego, niektóre zaś w okolicach zatopionego końca szklanej rurki, przez którą wypompowywane jest powietrze.

Drucik żarnika i siatka sterująca znajdują się między patrzącym a świecąca anodą. Muszą więc być tak małe, aby były niezauważalne. Jednocześnie drut żarnika powinien być rozgrzany do około 1000 stopni Celsjusza!

Sterowanie segmentami lub matrycą VFD jest zbliżone bardziej do

List. 4. Przykład programu obsługi wyświetlacza VFD w języku Bascom z wykorzystaniem portu szeregowego

```

$regfile = "8515DEF.DAT"
$baud = 4800 'ustawienie szybkości
'transmisji UART
$crystal = 7372800

Do
    Printbin &H1B; &H4C; 0 '30%
    Printbin &H0E 'kasowanie ekranu
    Printbin &H1B; &H48; 0 'ustawienie
        'kursora na początku ekranu 0,0
    Waitms 500
    Print "Noritake VFD display";
    Waitms 500
    Print "CU20045SCPB-T23A"
    Waitms 500
    Print "RS232:19200,n,8,1"
    Waitms 500
    Print "Bascom is ok!" 'wysyłamy
        'napis na ekran

    Waitms 800

    Printbin &H1B; &H4C; &H40 '50%
    Waitms 800
    Printbin &H1B; &H4C; &H80 '75%
    Waitms 800
    Printbin &H1B; &H4C; &HC0 'tutaj
        'regulacja jasności 100%
    Waitms 800
    Printbin &H1B; &H4C; &H80 '75%
    Waitms 800
    Printbin &H1B; &H4C; &H40 '50%
    Waitms 800
Loop

```

stosowanego dla wyświetlaczy LED niż LCD. Przeważnie nie musimy się jednak zajmować sterowaniem - nadzoruje je wbudowany przez producenta sterownik wyświetlacza. Wystarczy wiedzieć, że VFD może być przezeń sterowany zarówno statycznie - poprzez przyłożenie odpowiedniego napięcia - jak i dynamicznie - to znaczy z multipleksowaniem. Ze względu na bardzo dużą liczbę wyprowadzeń koniecznych przy zastosowaniu metody statycznej (**rys. 3**), przeważnie stosowane jest wyświetlanie dynamiczne (**rys. 4**). Przy takim wyświetlaniu niższa jest cena wyświetlacza i mniejsza złożoność.

Starsze modele wyświetlaczy VFD wymagały doprowadzenia wielu napięć sterujących. Wymagane było zarówno odpowiednie napięcie siatki, jak i anodowe oraz żarzenia. Skomplikowany sposób zasilania był przyczyną, że nie były one zbyt chętnie stosowane przez konstruktorów, choć można je było spotkać w różnych wyrobach przemysłowych, takich jak: kalkulatory stacjonarne, magnetowidy czy zegary cyfrowe. Charakterystyczną jest bowiem dla nich znakomita czytelność w różnych warunkach oświetlenia.

List. 5. Program do obsługi wyświetlacza VFD przez UART w języku asembler 8051

```

$include (REG_51.PDF)
NAME VFDTest
DSEG AT 20H
Status:DS 1
FlagaRXBIT Status.0
FlagaTXBIT Status.1
TXDone BIT Status.2

BuforRX: DS 1
BuforTX: DS 1

;wektor obsługi przerwania po reset
CODE AT 0H
    JMP Init
;wektor obsługi przerwania od SPI
CODE AT 23H
    JMP IrqSPI

CODE AT 30H
;początek programu głównego
;i wyprowadzenie napisu
VFD_Init: DB 1BH,49H,1BH,4CH,40H,0
Napis: DB 'Noritake VFD
ver.1,0 dd.2001/10/1SPI:9600,n,8,1',0
Init:
;ustawienie stosu
    MOV SP,#0E0H
    ACALL SPI_Init
    MOV B,#3
    MOV DPTR,#VFD_Init
    ACALL StringOut
    MOV DPTR,#Napis
    ACALL StringOut
    AJMP $

;*****
;Obsługa transmisji przez
SPI;*****
;obsługa przerwania od SPI
IrqSPI:JBC RI,RXIrq
;Czy to znak przychodzący?
TXIrq: JBC FlagaTX,SendIt
;Nie,wysyłaj dane
    CLR TI
    SETB TXDone
    JMP SPI_Ret
SendIt:MOV SBUF,BuforTX
    CLR TI
    CLR TXDone
    JMP SPI_Ret
RXIrq: MOV BuforRX,SBUF
;Tak,odbiór-czytaj znak
    SETB FlagaRX;Ustaw flagę odbioru
    SPI_Ret: RETI

;inicjalizacja UART
SPI_Init: CLR TR1
    CLR FlagaTX
    CLR FlagaRX
    SETB TXDone
    MOV SCON,#01010000B
    MOV TMOD,#00100001B
;timer 1 generuje "baude
;rate"
;timer 0 jako 16-bit timer
    MOV PCON,#0
;pojedyncza prędkość transmisji
    MOV TH1,#254
;th1 = 256-(11.0592e6/384x9600)
    SETB TR1
    SETB ES
    SETB EA
    RET

;Odczytuje znak i podaje go w A
CharIn:JNB FlagaRX,$
;Czekaj do momentu odbioru
    MOV A,BuforRX
    CLR FlagaRX
    RET

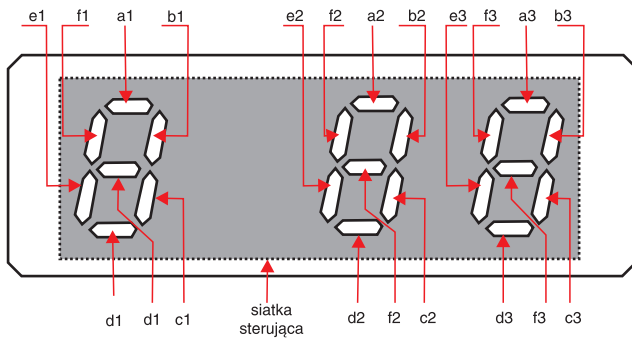
;Wyprowadza znak podany w A
CharOut: JB FlagaTX,$
;Nie za szybko, bo nastąpi blokada
    MOV BuforTX,A ;Wyślij znak
    SETB FlagaTX
    JNB TXDone,CharOut_Ret
    SETB TI
CharOut_Ret:
    RET

;Zwraca CY=0, jeśli znak nie jest
;"gotowy",CY=1 i znak w A jeśli wszystko ok
;Stan interfejsu SPI może być również
;sprawdzany poprzez bit RI
SPI_Status: MOV C,FlagarX
    JNC SPISta_Ret
    CALL CharIn
SPISta_Ret: RET

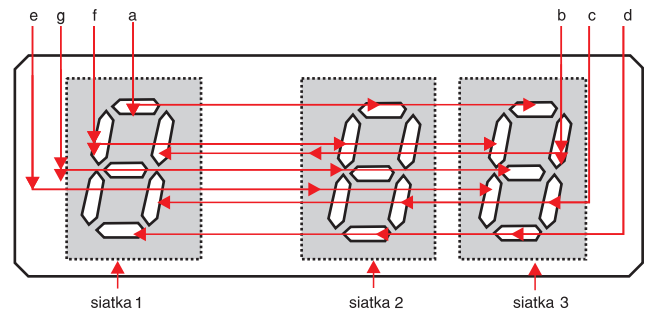
;Adres łańcucha do wysłania w DPTR,
;transmisja kończona jest przez znak 0x00.
StringOut: CLR A
    MOVC A,@A+DPTR
    CJNE A,#0,StrOut_1
    AJMP StrOut_2
StrOut_1: CALL CharOut
    INC DPTR
    JMP StringOut
StrOut_2: CLR A
    RET
END

```

**Nowoczesne wyświetlacze VFD są tak łatwe w stosowaniu jak popularne moduły LCD. Mają one podobny układ wyprowadzeń i są sterowane w taki sam sposób.**



Rys. 3. Połączenia segmentów w wyświetlaczu VFD sterowanym statycznie



Rys. 4. Połączenia segmentów w wyświetlaczu VFD sterowanym multipleksowo

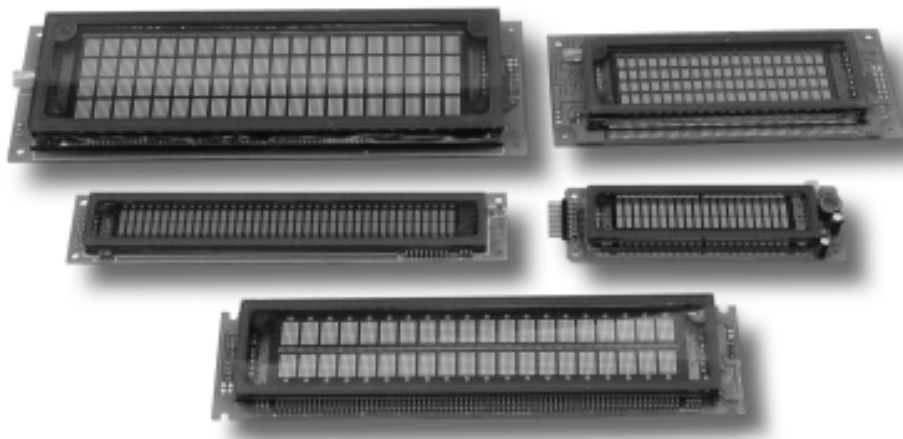
Obecnie najchętniej stosowane są te wyświetlacze VFD, które są zasilane z pojedynczego źródła napięcia i same wytwarzają niezbędne im do pracy napięcia.

**Przykłady programów sterujących**

Większość współcześnie produkowanych wyświetlaczy jest wyposażona w interfejs równoległy zgodny pod względem wyprowadzeń i realizowanych funkcji z popularnym sterownikiem HD44780. Można więc odłączyć wyświetlacz LCD wyposażony w interfejs zgodny z tym standardem, a w jego miejsce podłączyć równoważny

świetlacz LCD może być wykorzystywany przez producentów również innych modeli wyświetlaczy. W związku z tym programy sterujące pracą wyświetlacza LCD mogą być z powodzeniem użyte również dla VFD. Programy obsługi wyświetlaczy LCD były opisane w 3. i 4. odcinkach kursu programowania w języku C dla mikrokontrolerów z rodziny 8051 (EP7 i 8/2002). W przypadku języka Bascom również nie ma większych kłopotów: wystarczy znajomość kilku poleceń związanych z obsługą wyświetlania na LCD, takich jak: *Config Lcd*, *Config Lcdpin*, *Locate* itp. Ze znalezieniem przykładów

VFD: 2 linie po 20 znaków każda, doskonała jakość obrazu oraz tylko jedno napięcie niezbędne do jego zasilania. Pewną przeszkodą w jego wykorzystaniu jest specyficzny interfejs równoległy wymagający specjalnego sposobu sterowania, właściwego tylko temu modelowi wyświetlacza (na przykład sygnał BUSY wyprowadzony jest oddzielnie). Oczywiście możliwe jest napisanie programu sterującego, ale przy zmianie modelu wyświetlacza może się okazać, że konieczna będzie modyfikacja programu obsługi wyświetlania. Na szczęście producent wyposażył wyświetlacze w dwa rodzaje interfejsu: RS232, a raczej zgodny z jego specyfikacją transmisji, lecz pracujący z wykorzystaniem poziomów napięć TTL oraz równoległy. Wykorzystanie transmisji szeregowej nie wiąże się z żadnymi odstępstwami od standardu i eliminuje konieczność wykonania szeregu połączeń. Wyświetlacz wyposażony jest w trójstopniowe złącze, na którego wyprowadzenie 1 doprowadzane jest napięcie zasilania +5 V, na 2 sygnał danych, na 3 masa. Wykorzystywane jest wyłącznie wyprowadzenie TxD mikrokontrolera (transmisja zwrotna nie jest przeprowadzana). Nie ma potrzeby kontrolowania flagi zajętości oraz stanu wyświetlacza - wszystkim zajmuje się układ kontrolera. Należy tylko pamiętać o poprawnym ustawieniu parametrów transmisji. Opis sposobu wykonania niezbędnych nastaw można znaleźć w dokumentacji producenta. Osobiście bardzo mi się ta alternatywa podoba.



mu odpowiednik VFD (często nawet bez zmiany kolejności wyprowadzeń). Jedyna różnica polega na niewykorzystaniu przez VFD niezbędnego dla LCD napięcia regulacji kontrastu, ponieważ kontrast jest zawsze taki sam (jednakowo dobry) i regulować można tylko jasność świecenia znaków. Regulację tę przyprowadza się jednak nie za pomocą napięcia zewnętrznego, lecz programowo. Niewykorzystane będą również wyprowadzenia podłączenia napięcia podświetlenia tła.

Jak wynika z danych zawartych w tab. 1, sterownik HD44780 skonstruowany z przeznaczeniem dla wy-

takich programów przeznaczonych dla dowolnego modelu mikrokontrolera czy komputera PC nie powinno być większych trudności.

Trochę gorzej jest w przypadku starszych modeli wyświetlaczy, produkowanych gdy nie był jeszcze ustalony żaden standard sterowania i każdy z producentów budował własny interfejs. Pewnym ratunkiem może być wykorzystanie interfejsu szeregowego, w który wyposażane były niektóre z modeli wyświetlaczy, na przykład te produkowane przez firmę Noritake. Jako przykład niech posłuży wyświetlacz CU20025-T20A. Posiada on wszystkie cechy nowoczesnego

Na list. 3, 4 i 5 zamieszczono przykłady programów napisanych dla tego modelu wyświetlacza w językach Bascom, C i Asembler 51.

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

Dotatkowe materiały oraz oprogramowanie jest dostępne w Internecie pod adresem: <http://www.noritake-iron.com/Softview/softviewmain.htm>.