

# Alfanumeryczne wyświetlacze LCD

## część II

*Inteligentne wyświetlacze alfanumeryczne LCD są elementem coraz częściej spotykanym w sprzęcie powszechnego użytku: od urządzeń gospodarstwa domowego, poprzez systemy alarmowe, na medycznym sprzęcie profesjonalnym skończywszy. Dla wielu elektroników amatorów, chcących wykorzystać te efektowne elementy we własnych konstrukcjach są one często tylko marzeniem. W wielu przypadkach powodem takiego stanu rzeczy nie jest bynajmniej ich cena, lecz brak wiedzy o praktycznych sposobach na zmuszenie tego elementu do wyświetlania „tego co akurat w danej chwili chcemy...”*

*W niniejszym artykule autor w przystępny sposób stara się zapoznać Czytelników z tymi elementami budzącymi często podziw i westchnienie niejednego początkującego konstruktora.*



Jak wspomniałem wcześniej wszystko to i dużo więcej można uzyskać przy pomocy 11 linii sterujących. Zanim jednak zapoznasz się szczegółowo ze znaczeniem poszczególnych sygnałów, przyjrzyj się „maksymalnym” możliwościom modułu opartego o wspomniany standard oparty o sterownik HD44780.

Nazwy angielskie podaję nie bez powodu, bowiem będziemy się nimi czasami dla wygody posługiwać podczas omawiania poszczególnych funkcji oferowanych przez wyświetlacz.

### Jak sterować modułem

Przypatrzmy się teraz dokładniej wspomnianym wcześniej w artykule sygnałom sterującym. Zanim omówię ich znaczenie powinniśmy wiedzieć że układ sterowany jest napięciami TTL, poziomy logiczne sygnałów niskiego i wysokiego wynoszą odpowiednio: 0...0,8V oraz 2,4...5V, czyli odpowiadają w przybliżeniu poziomom w typowych układach CMOS przy zasilaniu napięciem +5V.

Oto skrócone znaczenie poszczególnych sygnałów sterujących modułem:

- **D0...D7**: osiem sygnałów przekazywania danych pomiędzy modułem LCD a światem zewnętrznym, czyli np. zewnętrznym układem sterującym. Fachowo sygnały te nazywa się „szyną danych”. Informacja może być przekazywana w obie strony, tak w stronę wyświetlacza LCD, kiedy to np. układ zewnętrzny chce wyświetlić jakiś znak, lub w drugą stronę. Ten przypadek wymaga dłuższego wyjaśnienia dlatego omówię go w dalszej części artykułu; Numeracja poszczególnych linii jest zgodna oczywiście ze standardem, czyli najmłodszy bit informacji to D0, najstarszy – D7.
- Sygnał **RW**: podanie niskiego poziomu na to wejście powoduje ustawienie modułu LCD w trybie odbioru informacji z układu sterującego (poprzez linie D0...D7), tryb ten nazywa się „trybem zapisu” – ang.

„Write”. Stan wysoki na tym wejściu ustawia moduł w tryb wysłania informacji do zewnętrznego urządzenia sterującego jego pracą. Dzięki temu ustalany jest kierunek komunikacji z wyświetlaczem LCD.

- Sygnał **RS**: podanie stanu wysokiego na to wejście, przez zewnętrzny układ sterujący, informuje moduł LCD o chęci przesłania danej do wyświetlenia (a ściślej do umieszczenia znaku w pamięci DD RAM wyświetlacza), lub odczyt tej pamięci z modułu do układu sterującego celem np. weryfikacji zapisanego wcześniej tekstu. Stan niski zaś informuje moduł o tym że układ sterujący chce przesłać instrukcję, dzięki której możliwe jest wywołanie wcześniej wspomnianych funkcji dodatkowych modułu takich jak: czyszczenie wyświetlacza, ustawianie kursora, itp. Tak dzieje się jeżeli przy tym sygnał  $R/W=0$ , czyli żądamy zapisu instrukcji do modułu. W przypadku gdy  $RS=0$ , i  $RW=1$  możliwe jest sprawdzenie „stanu zajętości” modułu, poprzez odczyt stanu na

linii D7. Jeżeli po tym sygnał D7 ma poziom wysoki, znaczy to że moduł wykonuje wewnętrzną operację i nie jest gotowy do odebrania kolejnej instrukcji z układu sterującego. W przypadku gdy przy odczycie linia D7 jest w stanie niskim oznacza to że moduł może odebrać kolejne polecenie od użytkownika. Znaczenie tego typu operacji wyjaśnię za chwilę.

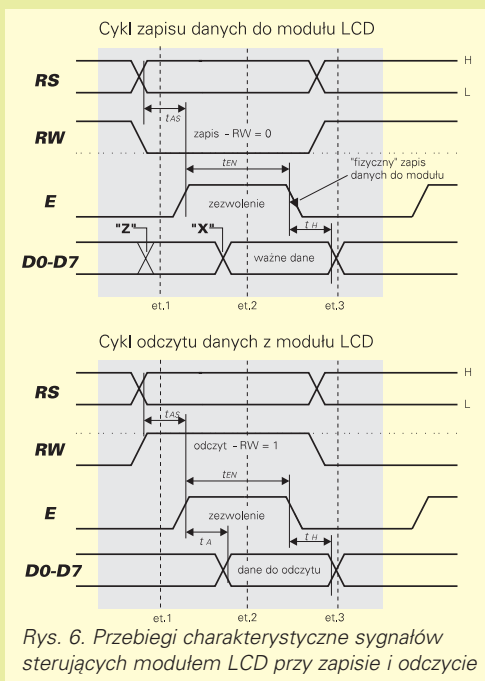
- Sygnał **E** (ang. „enable” - zezwolenie): podanie dodatniego impulsu na to wejście powoduje odebranie przez znajdujący się w module mikrosterownik HD44780, informacji z linii D0...D7 oraz RS i RW. W przypadku gdy sygnał  $RW=1$  (odczyt informacji z wyświetlacza LCD) podczas trwania tego impulsu na liniach D0...D7 pojawia się żądana informacja, dzięki czemu może być odczytana przez zewnętrzny układ sterujący. Zapis danej do wyświetlenia lub instrukcji ( $RW=0$ ) następuje przy opadającym zboczku sygnału E. Zbierzmy razem przytoczone tu informacje na temat sygnałów sterujących w tabeli 1.

#### Dane techniczne :

- a) wielkość bufora znaków (DD RAM): 80 znaków (80 bajtów)
- b) możliwość sterowania (czyli podawania kodów znaków) za pomocą interfejsu 8 lub 4-bitowego (czyli za pomocą 8 lub tylko 4 linii)
- c) zawartość generatora znaków CG ROM:
  - znaki 5 x 7 punktów: 160 znaków
  - znaki 5 x 10 punktów: 32 znaki
- d) możliwość odczytu wpisanych wcześniej znaków do DD RAM oraz CG RAM
- e) szeroka gama instrukcji pomocniczych:
  - czyszczenie wyświetlacza, (ang. „Clear Display”)
  - ustawienie kursora w pozycji początkowej (DD RAM = 0) (ang. „Cursor Home”)
  - włączenie / wyłączenie wyświetlacza (chodzi o samo pole odczytowe) (ang. Display ON/OFF”)
  - włączenie lub wyłączenia kursora (ang. „Cursor ON/OFF”)
  - możliwość zdefiniowania znaku lub kursora „migającego” (ang. „Cursor Blink”)
  - przesunięcie kursora: w lewo lub w prawo (ang. „Cursor Shift”)
  - przesunięcie całego tekstu: w lewo lub w prawo (ang. „Display Shift”)
- f) wbudowany układ automatycznego resetowania modułu po włączeniu napięcia zasilającego (ang. „Internal reset circuit”).

W zależności od kombinacji sygnałów RS i RW możliwe są przedstawione w niej operacje. Pamiętajmy przy tym że sygnał E jest tylko „zezwoeniem” dla modułu na odczyt stanów tych wyjść i podjęcie odpowiedniej operacji zgodnej z tabelą 1.

Jak zatem fizycznie sterować sygnałami RS, RW i E tak aby informacje podane na szynę D0...D7 zostały prawidłowo zaakceptowane przez moduł. Na **rysunku 6** przedstawiono przybliżone zależności czasowe przy generacji tych sygnałów, tak przy operacji odczytu informacji z modułu LCD jak przy zapisie.



Rys. 6. Przebiegi charakterystyczne sygnałów sterujących modulem LCD przy zapisie i odczycie

Wyjaśnijmy sobie dokładniej to co pokazano na rysunku. Aby uprościć analizę zaznaczyłem trzy hipotetyczne „etapy”, oznaczone jako et.1, et.2 i et.3. Odpowiadają one trzem operacjom, jakie powinien przewidzieć konstruktor układu, w którym wykorzystywany jest moduł LCD. Rozpocznijmy od zapisu instrukcji lub danej, wykres zapisu odnosi się do obu tych przypadków,

z tą różnicą że sygnał RS przyjmuje raz wartość „0”, raz „1”, zgodnie z tabelą 1.

et.1 : na początku należy ustawić odpowiedni poziom na linii RS, a na linii RW powinno być ustawione oczywiście „0”, bo dokonujemy operacji zapisu do modułu. Należy także podać na linii D0...D7 kod danej (gdy RS=0) lub instrukcji (gdy RS=1) zgodnie z tabelą 2. Linia E powinna w tym czasie pozostawać w stanie „0”. Najlepiej jest to wykonać w momencie oznaczonym na wykresie jako „Z”, czyli przy okazji ustawiania sygnałów RS i RW. W katalogach producenta modułów Hitachi ustalenie się szyny danych D0...D7 może być opóźnione i nastąpić dopiero w momencie kiedy sygnał E ma już poziom wysoki (punkt „X”), jednak ja radzę zrobić to wcześniej zgodnie z rysunkiem.

et.2 : następnie należy podać impuls o czasie trwania minimum  $t_{EN}$ , podczas opadającego zbocza tego impulsu dane z szyny D0...D7 zostają „fizycznie” odebrane przez moduł LCD. Po tym zboczach, które kończy cykl właściwego zapisu do modułu, powinno się odczekać czas  $t_H$ , podtrzymując dane na liniach D0...D7.

et.3 : sygnał E przyjmuje ponownie stan „0”, mija czas podtrzymania. Stan linii RS, RW oraz szyny danych jest wtedy nieistotny. Zapis został zakończony. Moduł wykonuje teraz wewnętrzną operację przez czas zależny od rodzaju wpisanej instrukcji lub danej, a jego wartości podane są w tabeli 2. Następnym cyklem zapisu (odczytu) może się rozpocząć po odczekaniu tego czasu, lub sprawdzeniu stanu „flagi zajętości” poprzez odczyt informacji z modułu LCD.

Przebieg odczytu przedstawia druga połowa rysunku 6.

W tym przypadku postępujemy podobnie jak przy zapisie, czyli na początku ustalamy poziom sygnału na linii RS, linia RW powinna się znaleźć w stanie wysokim (RW=1: odczyt). Następnie zmieniamy stan linii E na wysoki. Po czasie  $t_A$  dane z modułu pojawiają się na liniach D0...D7, wtedy mogą być odczytane przez zewnętrzny

układ sterujący użytkownika. Po odczytaniu należy zakończyć procedurę odczytu podając na linii E poziom niski. Kończy to cykl, po tym poziomy na liniach RS i RW są nieistotne. Kolejny cykl może się zacząć po czasie 1 $\mu$ s (zgodnie z tabelą 2).

Minimalne czasy trwania przedstawiono na rysunku 6 oraz ich znaczenie jest następujące:

- $t_{AS}$ : czas od ustawienia sygnałów RS i RW do uaktywnienia sygnału E, min.: 140ns;
- $t_{EN}$ : czas trwania impulsu E, min.: 450ns;
- $t_H$ : czas podtrzymania sygnałów RS, RW oraz danej po opadającym zboczach sygnału E, min.: 20ns;
- $t_A$ : przy odczycie: czas od momentu uaktywnienia sygnału E do pojawienia się informacji na szynie danych, maks. 320ns.

Jak widać, poszczególne czasy są bardzo krótkie, warto jednak o nich pamiętać przy budowaniu układu sterującego pracą modułu LCD.

Warto też zwrócić uwagę (rysunek 6) że zapis informacji do modułu następuje fizycznie podczas opadającego zbocza sygnału E, natomiast odczyt jest możliwy podczas trwania wysokiego stanu sygnału E.

Zapis i odczyt informacji z modułu może odbywać się dzięki tym samym liniom danych D0...D7 dzięki temu, że są one liniami „trójstanowymi”. Czyli w przypadku zapisu linie te działają jako wejścia informacji (wewnętrzne wyjścia są w stanie wysokiej impedancji), a w przypadku odczytu, po podaniu poziomu „1” na linię E trójstanowe wyjścia zostają odblokowane i dzięki temu moduł LCD może przekazać informację na końcówki D0...D7.

I choć tekstowe moduły LCD przeznaczane są głównie do współpracy w układach wykorzystujących mikroprocesory, to można je także stosować w prostych sterownikach „niemikroprocesorowych”. Przykład takiego rozwiązania z wykorzystaniem zwyczajnej pamięci EPROM to opublikowany w poprzednim numerze EdW „najprostszy sterownik wyświetlacza LCD” kit AVT-2251 – zapraszam do lektury. Najczęściej w takich prostych układach sterujących modułami LCD nie jest wykorzystywany tryb odczytu danych czyli: „flagi zajętości” oraz adresu w DD RAM lub CG RAM. Ponieważ moduł LCD wykonuje każdą wewnętrzną operację przez określony (tabela 2) maksymalny czas, nie jest w zasadzie potrzebne sprawdzanie tej flagi. Wystarczy przecież odczekać z małym zapasem czas podany w tabeli 2, co gwarantuje że następny rozkaz z układu sterującego pracą modułu LCD zostanie prawidłowo odebrany. Dlatego w wielu aplikacjach wyświetlaczy LCD, także mikroprocesorowych końcówka RW modułu jest na stałe zwarta do masy. Układ nadrzędny zajmuje się jedynie sterowaniem sygnałów RS i E oraz oczywiście podawaniem informacji na szynę danych D0...D7. W efekcie upraszcza to znacznie obsługę wyświetlacza, nie ujmując mu jego funkcjonalności.

W prawdziwych „rasowych” zastosowaniach wykorzystujących mikrokontrolery funkcja odczytu danych ma jednak zastoso-

Tabela 1

RS	RW	Działanie modułu
0	0	zapis instrukcji (rozkazu) do modułu przez zewnętrzny układ sterujący. Kod instrukcji podawany jest na linii D0...D7, lista instrukcji znajduje się w tabeli 2.
1	0	zapis danej do pamięci DD RAM (lub do CG RAM), jeżeli wskaźnik adresu w DD RAM znajduje się w obszarze „okna wyświetlania” następuje wyświetlenie znaku na displeju LCD. Kod znaku podawany jest na linii D0...D7, zgodnie z tabelą na rysunku 4. Zapis do CG RAM używany jest w przypadku definiowania własnego znaku przez użytkownika.
0	1	odczyt tzw. „flagi zajętości” modułu – bit D7, oraz bieżącej pozycji wskaźnika adresu w DD RAM (lub w CG RAM) – bity D6...D0
1	1	odczyt danej z DD RAM (lub z CG RAM) z pozycji którą wskazuje bieżąca zawartość wspomnianego wskaźnika adresu.

## Podręczny poradnik elektronika

wanie – jest po prostu w pewnych względach praktyczna w takim układzie pracy.

### Funkcje kontrolne wyświetlacza

Ponieważ podałem wcześniej na rysunku 4 zestaw kodów odpowiadających wyświetlanym przez moduł znaków pora zapoznać się z listą i znaczeniem poszczególnych instrukcji sterowania wyświetlaczem. W tabeli 2 ujęto wszystkie polecenia wyświetlacza, te służące zarówno do odczytu jak i do zapisu. Jeżeli chcesz w przyszłości jeżeli zaznajomisz się i wykorzystasz opisywane tu moduły w praktyce, napotykając jakiś nietypowy jego rodzaj, będzie mógł łatwo znaleźć analogie instrukcji w często angielskojęzycznej dokumentacji do-

łączanej przez sprzedawcę lub producenta do oferowanego typu wyświetlacza. Jeżeli nie znasz angielskiego lub go nie lubisz, nie przejmuj się, w przedostatniej kolumnie znajduje się krótki opis danej instrukcji. W kolumnach RS, RW D7...D0 podane są kombinacje poszczególnych sygnałów sterujących i szyny danych, które powinny być ustawione przez uaktywnienie sygnału E, jak opisałem wcześniej. W niektórych kratkach tabeli występują literki np.: dla instrukcji: „Display ON/OFF” są to : D, C, i B. Oznacza to że w zależności od efektu jaki chcemy uzyskać na wyświetlaczu, należy te bity (pozycje) wyzerować lub ustawić zgodnie z opisem znajdującym się pod koniec tabeli. Wyjaśnione są tam wszystkie symbole występujące w tabelce. I tak np. jeżeli chcesz: włączyć wyświetlacz i pokazać niemigający kursor powinienes podstawić następujące wartości:

D=1, C=1, B=0, czyli w efekcie informacja podana na szynę danych przez zapie-

sem instrukcji (RW=0, RS=0) powinna mieć postać: 00001110 binarnie, prawda że proste.

Poniżej przedstawię krótki opis poszczególnych instrukcji i efekt ich wykonania.

(1) „Clear display” – czyszczenie wyświetlacza

RS	R/W	D7..D0								
0	0	0	0	0	0	0	0	0	0	1

Całą pamięć DD RAM zostaje wypełniona spacjami (20h), wskaźnik (kursor) adresu DD RAM zostaje wyzerowany. (=0). Jeżeli wyświetlacz był „przesunięty”, wraca na swoje miejsce. Rozkazu ustawia bit I/D w słowie „Entry Mode”. Bit „S” w tym słowie nie zmienia się.

(2) „Return home” – ustawienie kursora na poz. początkowej

RS	R/W	D7..D0								
0	0	0	0	0	0	0	0	0	1	*

Tabela 2

Instrukcja	Dane										Opis	Czas wykonania max.
	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0		
Clear display	0	0	0	0	0	0	0	0	0	1	Czyści wyświetlacz i ustawia kursor na początku (adres=0)	120 µs... 4,9 ms
Return home	0	0	0	0	0	0	0	0	1	*	Ustawia kursor na pozycji początkowej (adres=0). Jeżeli napis był przesunięty, ustawia go na pozycji oryginalnej. DD RAM nie ulega zmianie.	120 µs... 4,8 ms
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Ustawia kierunek przesuwu kursora i włącza/wyłącza funkcję przesuwu napisu przy zapisie do DD RAM.	120 µs
Display ON/OFF	0	0	0	0	0	0	1	D	C	B	Włącza/wyłącza wyświetlacz (D), kursor (C) i funkcję migania kursora (B) - „blink”.	120 µs
Cursor & display shift	0	0	0	0	0	1	S/C	R/L	*	*	Przesuwa kursor i napis bez zmiany zawartości DD RAM.	120 µs
Function set	0	0	0	0	1	DL	N	F	*	*	Ustala ilość bitów danych interfejsu (DL) liczbę wyświetlanych linii (L), oraz format znaku (F)	120 µs
Set CG RAM address	0	0	0	1	A <sub>CG</sub>						Ustawia adres w CG RAM. Wszystkie operacje zapisu danej odnoszą się do tej pamięci po wywołaniu tej instrukcji.	120 µs
Set DD RAM address	0	0	1	A <sub>DD</sub>						Ustawia adres w DD RAM. Wszystkie operacje zapisu danej odnoszą się do tej pamięci po wywołaniu tej instrukcji	120 µs	
Read busy flag	0	1	BF	AC						Odczytuje stan flagi zajętości „Busy Flag” oraz bieżący adres w pamięci DD RAM lub CG RAM	1 µs	
Write data to CG or DD RAM	1	0	Dana do zapisu								Zapisuje daną (znak) do pamięci DD RAM.	120 µs
Read data from CG or DD RAM	1	1	Odczytana dana								Odczytuje daną (znak) z pamięci DD lub CG RAM wyświetlacza	120 µs
	I/D = 1: Zwiększenie (+1), I/D = 0: Zmniejszenie (-1) S = 1: Towarzyszy przesuwaniu napisu. S/C = 1: Przesuwanie napisu, S/C = 0: Przesuwanie kursora R/L = 1: Przesuwanie w prawo. R/L = 0: Przesuwanie w lewo. DL = 1: Interfejs 8-bitowy, DL = 0: Interfejs 4-bitowy N = 1: 2 linie N = 0: 1 linia F = 1: znaki 5 x 10 punktów F = 0 : znaki 5 x 7 punktów BF = 1: Moduł zajęty BF = 0: Gotowy na następną instrukcję Symbolem „*” (gwiazdki) oznaczono bity nie mające znaczenia.										DD RAM: Pamięć znaków (wyświetlania) CG RAM: Pamięć generatora znaków A <sub>CG</sub> : Adres w pamięci CG RAM A <sub>DD</sub> : Adres w pamięci DD RAM AC : licznik (wskaźnik) adresu w pamięci DD lub CG RAM	Czas wykonania może być mniejszy w zależności od częstotliwości zegara wbudowanego w moduł LCD.

Zeruje wskaźnik adresu DD RAM, kursor zostaje przesunięty do pozycji 0. Przesunięty tekst powraca na swoje miejsce, zawartość pamięci DD RAM nie ulega zmianie.

(3) „Entry mode set” – sposób sterowania wyświetlaczem

RS	R/W	D7..D0							I/D	S	
0	0	0	0	0	0	0	0	0	1	I/D	S

I/D: inkrementuje (I/D=1) lub dekrementuje (I/D=0) wskaźnik adresu DD RAM o 1 po każdorazowym zapisie znaku do tej pamięci. Kursor zostaje przesunięty w prawo gdy I/D=1, lub w lewo gdy I/D=0. To samo dotyczy pamięci CG RAM przy zapisie matrycy znaku użytkownika.

S: powoduje przesunięcie całej zawartości DD RAM (napisu) w lewo lub prawo w zależności od bitu I/D. W praktyce wygląda to tak jakby kursor stał w miejscu, a cały napis przesuwał się. Zapis do CG RAM przy S=1 nie powoduje przesunięcia się napisu.

(4) „Display ON/OFF” – włączanie wyświetlacza i kursora

RS	R/W	D7..D0							D	C	B
0	0	0	0	0	0	0	0	1	D	C	B

D: włącza wyświetlacz gdy D=1, i gasi gdy D=0. Zmiana tego bitu nie powoduje zmiany zawartości DD RAM (wprowadzonego tekstu)

C: pokazuje kursor gdy C=1 i chowa gdy C=0. Nawet gdy kursor zostanie ukryty, jest nadal aktywny i podąża wraz ze wskaźnikiem pamięci DD RAM podczas operacji zapisu.

B: po ustawieniu tego bitu (B=1) kursor zajmuje całą matrycę znaku, dodatkowo migocząc na przemian ze znajdującym się „za nim”, wyświetlanym znakiem.

Na **rysunku 7** pokazałem możliwe kształty kursora oraz sposób wyświetlania migoczącego znaku.

(5) „Cursor & display shift” – kontrola kursora i przesuwania tekstu.

RS	R/W	D7..D0							S/C	R/L	*	*
0	0	0	0	0	0	0	0	1	S/C	R/L	*	*

Przesuwa kursor lub napis w prawo lub lewo bez zmiany zawartości pamięci DD RAM. W 2-liniowym trybie pracy kursor przechodzi do drugiej linii w momencie minięcia

40 pozycji w pierwszej linii. W tym trybie przy przesuwaniu napisu, obie linie są przesuwane jednocześnie, tzn. że np. ostatni znak w 1 linii trafia na miejsce pierwsze w tej samej linii, a nie przechodzi do linii drugiej. To samo dotyczy linii nr 2. W praktyce wygląda to jak przesuwanie poziome dwóch niezależnych napisów w 2 liniach. W **tabeli 3** zestawiono działanie kombinacji bitów S/C i R/L.

Tabela 3

S/C	R/L	Efekt
0	0	Przesuwa kursor w lewo, wskaźnik adresu zostaje zmniejszony o 1
0	1	Przesuwa kursor w prawo, wskaźnik adresu zostaje zwiększony o 1
1	0	Przesuwa cały napis w lewo, kursor podąża za przesuwanym tekstem
1	1	Przesuwa cały napis w prawo, kursor podąża za przesuwanym tekstem

(6) „Function Set” – ustawienie funkcji dodatkowych

RS	R/W	D7..D0							DL	N	F	*	*
0	0	0	0	0	0	0	1	DL	N	F	*	*	

DL: ustala szerokość magistrali danych. Gdy DL=1 dane przesyłane są w postaci 8-bitowej linia D0..D7. Kiedy DL=0, transmisja jest 4-bitowa: linie D4..D7. Gdy wybrany jest interfejs 4-bitowy każda dana lub rozkaz musi być przesłana w 2 cyklach, najpierw starsza część bajtu potem młodsza. Po każdej operacji należy sprawdzić „Busy Flag” lub odczekać czas określony w tabeli 1.

N: ustala tryb pracy 1-liniowy (N=0), lub 2-liniowy (N=1). Gdy aktywny jest tryb 1-liniowy, a niektóre z modułów mają fizycznie (SW1) ustawiony adres drugiej linii, pozostają nieużywane.

F: ustala rozmiar matrycy znaku; F=0 matryca ma 5 x 7 punktów, F=1 matryca 5 x 10. Nie wszystkie moduły LCD wykorzystują tę drugą możliwość, jej dość niktłe znaczenie opisywałem wcześniej w artykule.

Uwaga: w przypadku kiedy F=1 (5x10) oraz N=1 (tryb pracy 2-liniowy) nie jest możliwe wyświetlenie tekstu w dwóch liniach, w przypadku wyświetlaczy wieloliniowych, a jedynie w jednej.

(7) „Set CG RAM Address” – ustawienie adresu pamięci znaków użytkownika

RS	R/W	D7..D0							
0	0	0	1	A	A	A	A	A	A

Ustala adres aktualnego zapisu do pamięci matrycy znaku użytkownika CG RAM.

Dozwolony adres: 00h..3Fh jak podano w tabeli 3. Po tej operacji dane będą umieszczane od ustawionego adresu w CG RAM. (AAAAAA – 6 bitowy adres, zakres: 0..63)

(8) „Set DD RAM Address” – ustawienie adresu pamięci tekstu (wskaźnika)

RS	R/W	D7..D0							
0	0	1	A	A	A	A	A	A	A

Ustala adres aktualnego zapisu do pamięci tekstu DD RAM. Po tej operacji dane są umieszczane od ustawionego adresu w DD RAM. (AAAAAAA-7-bitowy adres)

Gdy N=0 (tryb 1-liniowy) dozwolony zakres adresu: 00h..27h, gdy N=1 (tryb 2-liniowy) adresy 1 linii: 00h..27h, 2 linii: 40h..67h

(9) „Read busy flag” and AC address – odczyt flagi zajętości

RS	R/W	D7..D0							
0	1	BF	A	A	A	A	A	A	A

Odczytanie stanu flagi zajętości „Busy Flag” oraz bieżącego adresu w CG lub DD RAM. Gdy po odczycie BF=1 znaczy to że moduł wykonuje wewnętrzną operację i nie przyjmie danej ani instrukcji. Następne dane powinny być przesyłane do wyświetlacza gdy BF=0.

AAAAAAA to 7-bitowy adres bieżącej pozycji w CG lub DD RAM, uwagi co do zakresów liczbowych adresu zgodne z punktem poprzednim.

(10) „Write data to CG or DD RAM” – zapis danej do CG RAM lub DD RAM

RS	R/W	D7..D0							
1	0	D	D	D	D	D	D	D	D

Wpisuje 8-bitową daną DDDDDDDD do pamięci tekstu DD RAM lub generatora znaków użytkownika CG RAM. To do jakiej pamięci zostaje zapisana dana zależy od tego, do jakiej pamięci odnosi się ostatnie ustawienie adresu, patrz instrukcje „Set CG RAM address” i „Set DD RAM address”. Po zapisie do pamięci DD RAM lub CG RAM wskaźnik adresu zostaje automatycznie inkrementowany lub dekrementowany o 1 w zależności od ustawienia polecenia „Entry Mode”.

Przykładowa kolejność instrukcji w celu zapisania 7-znakowego tekstu np.: „DISPLAYJ” pokazany jest w tabeli 4.

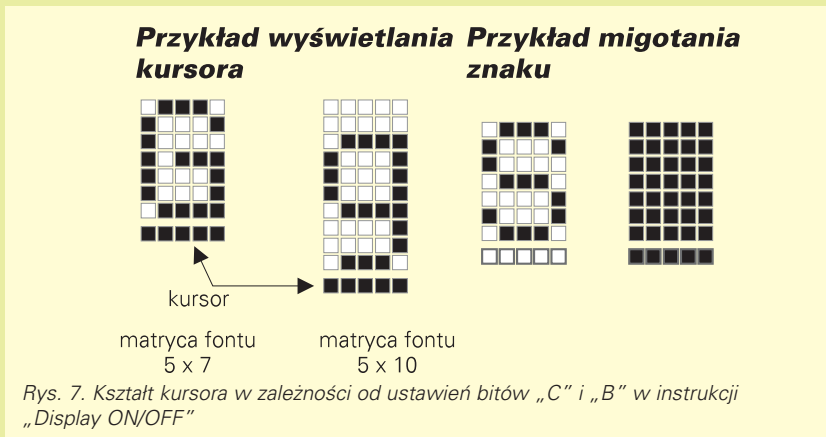
(11) „Read data from CG or DD RAM” – odczyt danej z CG RAM lub DD RAM

RS	R/W	D7..D0							
1	1	D	D	D	D	D	D	D	D

Odczytuje 8-bitową daną DDDDDDDD z modułu, zapisaną pod aktualnym adresem który wskazuje licznik adresu (odczytywany w sposób podany w pkt.9)

ciąg dalszy w następnym numerze.

Sławomir Surowiński



Rys. 7. Kształt kursora w zależności od ustawień bitów „C” i „B” w instrukcji „Display ON/OFF”