

INTELIGENTNE WYŚWIETLACZE



Rozwój technologii układów scalonych spowodował, że ceny układów scalonych bardzo się obniżyły. Dotyczy to także sprzętu mikroprocesorowego. Spadek cen spowodował wzrost zainteresowania tą technologią także i wśród majsterkowiczów, co zresztą dało się zauważyć w listach od Czytelników.

Większość urządzeń musi mieć jakiś element umożliwiający komunikację z otoczeniem — i jednym z takich elementów jest wyświetlacz. Okazuje się, że i tu technologia poszła naprzód, że proste, siedmiosegmentowe wyświetlacze, mogące wyświetlać tylko cyfry, należą już do przeszłości. Na naszym rynku można kupić tzw. *inteligentne wyświetlacze*, skonstruowane w technice ciekłokrystalicznej (LCD), dające możliwość wyświetlenia znaków złożonych z mozaiki punktów, wyposażone w zintegrowane układy sterowania. Cena takiego wyświetlacza w najprostszej wersji nie przekracza obecnie (lipiec 1993) 250 tys. zł, co, w stosunku do jego możliwości, wydaje się być ceną atrakcyjną.

Oferta krajowych handlowców obejmuje rozmaite wersje tych wyświetlaczy, jednak brak szczegółowej informacji o nich powoduje, że amatorzy majsterkowicze odnoszą się do nich niechętnie. Dlatego dziś piszemy o tym, jak podłączyć taki wyświetlacz do układu oraz... co z nim dalej robić, aby wyświetlił, co chcemy.

Opis struktury modułów wyświetlających

Wyświetlacze tej serii zawierają typowe układy scalone *dekodera* i *sterownika* (HD 44100 i HD 44780), różnią się natomiast modułem ciekłokrystalicznej matrycy, co limituje możliwą jednocześnie do wyświetlenia liczbę znaków i ich jakość. Przykładowo, wyświetlacz HDM-16108H-7 może wyświetlić naraz 16 znaków (uszeregowanych w jednej linii), z których każdy tworzy matrycę punktów $5 \times 7 + 1$. Ten stosunkowo dziwny zapis rozdzielczości spowodowany jest tym, że najniższy wiersz matrycy jest lekko odsunięty od pozostałych; pełni on rolę kursora, ale może być wykorzystywany jako część składowa wyświetlanego znaku. Spotykane poza tym rozdzielczości to: 5×8 i 5×10 punktów.

Wewnątrz wyświetlacza znajduje się pamięć o pojemności 80 znaków. Programujący wyświetlacz ma dostęp do całej tej pamięci — widzi on ją jako jedną, długą linię, której początek łączy się z końcem. Niejako nad tą linią przesuwa się okienko wyświetlacza — w przypadku modułu HDM-16108H-7 z linii tej możemy wybrać dowolny szesnastoznakowy „kawałek” przeznaczony do wyświetlenia.

Ponieważ początek linii łączy się z jej końcem, więc w zasadzie żaden ze znaków nie jest uprzywilejowany, jednak (ponieważ musimy je jakoś rozróżnić) ponumerujemy je w zakresie 0-79. Znak o numerze 0 będzie pierwszym znakiem w linii.

Wbudowany w moduł generator znaków zawiera 192 wbudowane wzorce (niektóre bardzo wymyślne). Kody liter (małych i dużych), cyfr oraz niektórych znaków specjalnych są zgodne z kodami ASCII. Należy zwrócić uwagę, że numeracja poszczególnych znaków jest nieciągła.

Dodatkowo użytkownik ma możliwość zdefiniowania 8 znaków o kształtach zaprojektowanych przez siebie. Przy pracy w rozdzielczościach $5 \times 7 + 1$ i 5×8 każdy znak definiuje się jako osiem kolejnych bajtów. Znaki definiowane przez użytkownika przyjmują kody 0-7 oraz dodatkowo są powielone dla lokacji 8-15. Sposób odwzorowania mapy znaku w ciąg bajtów i umiejscowienia tego ciągu w pamięci przedstawiony jest na rys. 1 na przykładzie litery „a”.

Sterownik dopuszcza sterowanie wyświetlaczy dwulinowych. Zrealizowane jest to tak, jakby każda linia była oddzielnym oknem, jednakże odległość między tymi oknami jest stała (równa 40 znakom). Długość i sposób ustawienia tych okien zależą od producenta: spotykane są wyświetlacze mające 2 linie po 32 znaki, ale są i takie, w których dwie linie po 8 znaków ustawione są „jedna za drugą” tworząc szesnastoznakowy pasek.

Jak już mówiliśmy, moduł wyświetlacza zawiera w sobie dwa rodzaje pamięci RAM:

1. Pamięć znaków (tzw. DD RAM, ang. *Display Data RAM*), w której przechowywane są kody wyświetlanych znaków; pamięć ta ma adresy w zakresie 0-80 i jest odwzorowywana na ekranie wyświetlacza.

2. Pamięć generatora znaków (tzw. CG RAM, ang. *Character Generator RAM*), w której umieszczone są mapy bitowe znaków definiowanych przez użytkownika; pamięć ta ma adresy w zakresie 0-64.

Pamięci te są całkowicie rozdzielne.

Dostęp do pamięci odbywa się w sposób pośredni, tzn. użytkownik nie ma dostępu do całego obszaru pamięci jednocześnie. Postrzega on moduł wyświetlacza w postaci rejestrów: rejestr instrukcji i rejestru danych. Rejestr danych jest „okienkiem”, które można ustawić tak, aby „widoczny w nim był” wybrany bajt z pamięci (DD RAM lub CG RAM), a następnie, poprzez to „okienko”, dokonuje się operacji na tym bajcie. Do ustawiania pozycji „okienka” (i wydawania wielu innych rozkazów) służy rejestr instrukcji.

Wpisanie bajtu do pamięci DD RAM spowoduje pojawienie się na ekranie kodowego oznaczenia tego bajtu (oczywiście, o ile okno ekranu obejmuje ten bajt swoim zasięgiem). Wpisanie bajtu do pamięci CG RAM spowoduje zmianę kształtu jednego ze znaków definiowanych przez użytkownika.

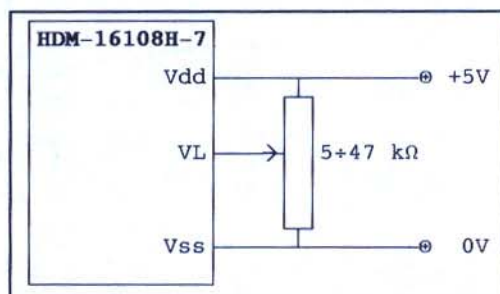
Aktualna pozycja „okienka” określona jest adresem bajtu w pamięci (CG RAM lub DD RAM), który ono wskazuje. W przypadku pamięci DD RAM będzie to jednocześnie pozycja kursora.

Ponieważ istnieje również możliwość odczytu zawartości pamięci (obu), więc w zasadzie pamięć tą niczym nie różni się od pamięci systemowej. Niektórzy wytwórcy sugerują nawet, że można nie używane komórki wyświetlacza wykorzystać jako dodatkową pamięć – być może, w pewnych specyficznych sytuacjach się to opłaca. Jeżeli jednak zdecydować się na to (a w zasadzie nie tylko wtedy), należy zdawać sobie sprawę z automatycznej zmiany pozycji wskaźników.

Rys. 1. Sposób definicji znaku przez użytkownika

Znak	o kodzie n	adres	bajt
□ □ □ □ □	n*8	00000000	
□ □ □ □ □	n*8+1	00000000	
□ ■ ■ ■ □	n*8+2	00001110	
□ □ □ □ ■	n*8+3	00000001	
□ ■ ■ ■ ■	n*8+4	00001111	
■ □ □ □ ■	n*8+5	00010001	
□ ■ ■ ■ ■	n*8+6	00001111	
□ □ □ □ ■	n*8+7	00000001	

Otóż, aby ułatwić wpisywanie znaków do wyświetlacza, wbudowano weń mechanizm autoinkrementacji/autodekrementacji. Oznacza to, że po wpisaniu znaku do wyświetlacza kursor automatycznie przesuwa się o jedną pozycję (w lewo lub w prawo, w zależności od nastaw). To samo zdarzy się, gdy odczytamy znak z wyświetlacza. (Przy dostępie do pamięci CG RAM obowiązuje ten sam mechanizm, z tym, że przesunięciu ulega nie kursor, lecz wskaźnik bajtu.)



Rys. 2. Sposób zasilania modułu wyświetlacza

Opis wyprowadzeń modułów

Moduł wyświetlacza komunikuje się z otoczeniem przez 14 wyprowadzeń, z których 3 służą do zasilania modułu, a 11 stanowi wyprowadzenia sygnałowe. Wszystkie wyprowadzenia sygnałowe dostosowane są do standardu TTL i charakteryzują się następującymi parametrami:

- Napięcie wejściowe zera logicznego 0 ÷ 0,6V,
- Napięcie wejściowe jedynki logicznej 2,4V ÷ 5V,
- Maksymalny prąd wyjścia w stanie zera logicznego 1,2 mA,
- Maksymalny prąd wyjścia w stanie jedynki logicznej 0,8 mA.

Dokładny opis wyprowadzeń modułu podany jest poniżej.

V_{SS} – masa układu.

V_{DD} – napięcie zasilania. Typowa wartość napięcia zasilania wynosi 5V, zasilanie napięciem powyżej 7V może spowodować uszkodzenie modułu. Prąd zasilania modułu nie przekracza 1,8mA.

V_L – napięcie regulacji kontrastu wyświetlacza. Maksymalne napięcie, które może być przyłożone na to wyprowadzenie, nie może przekroczyć 13,5V względem V_{SS}, jednak praktyka wykazuje, że jego wartość niezbędna do prawidłowej nastawy kontrastu jest mniejsza niż 5V i uzyskuje się ją z dzielnika napięcia zasilania (rys. 2).

RS – selektor rodzaju danych. Podczas wpisywania danych do modułu wyświetlacza stan tego wejścia

Higher Lower 4bit 4bit	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)												
xxxx0001	(2)												
xxxx0010	(3)												
xxxx0011	(4)												
xxxx0100	(5)												
xxxx0101	(6)												
xxxx0110	(7)												
xxxx0111	(8)												
xxxx1000	(1)												
xxxx1001	(2)												
xxxx1010	(3)												
xxxx1011	(4)												
xxxx1100	(5)												
xxxx1101	(6)												
xxxx1110	(7)												
xxxx1111	(8)												



określa, czy wpisywany bajt jest daną (gdy $RS=1$), czy też instrukcją sterującą (gdy $RS=0$). Innymi słowy, stan tego wyprowadzenia określa, czy operacja dokonywana jest na rejestrze danych, czy też rejestrze instrukcji.

R/W – określa kierunek przesyłu na szynie danych. Gdy $R/W = 0$, wówczas wyprowadzenia szyny danych są wejściami (moduł oczekuje danej), gdy $R/W = 1$, wyprowadzenia szyny danych są wyjściami (moduł wysyła daną).

E – sygnał wyzwalania. Sygnał jest aktywny poziomem 1, tzn. w tym czasie następuje transfer danych pomiędzy modułem wyświetlacza, a urządzeniem zewnętrznym. Jeżeli był to wpis do rejestru instrukcji, to w chwili przejścia $1 \rightarrow 0$ moduł wyświetlacza przechodzi do wykonania komendy.

D0-D7 – dwukierunkowa szyna danych (kierunek przesyłania danych określa stan linii R/W). Może ona pracować także w trybie pracy 4-bitowym; wówczas przesyłanie danych odbywa się w dwóch porcjach po szynach D7-D4, najpierw starszy półbajt, następnie młodszy.

Niekiedy moduł wyświetlacza wyposażony jest dodatkowo w elementy podświetlające ekran (diody elektroluminescencyjne), wówczas liczba wyprowadzeń zwiększa się o dwa.

Rozkład wyprowadzeń dla wyświetlaczy firm HANDOK, SHARP i DATAMEGA jest identyczny:

V_{ss}	V_{dd}	V_L	RS	R/W	E	D0-D7
1	2	3	4	5	6	7-14

(w przypadku istnienia diod podświetlających, ich zasilanie to wyprowadzenia 15 i 16).

Spotykane są również wersje wyświetlaczy o czterech liniach. W rzeczywistości jednak w obudowie modułu znajdują się dwa układy sterowników, z których każdy obsługuje dwie linie. Wyprowadzenia obu sterowników są połączone równolegle – z wyjątkiem sygnału zezwolenia E, które wyprowadzone są oddzielnie.

Dariusz Adam Przygoda

Dokończenie w następnym numerze

