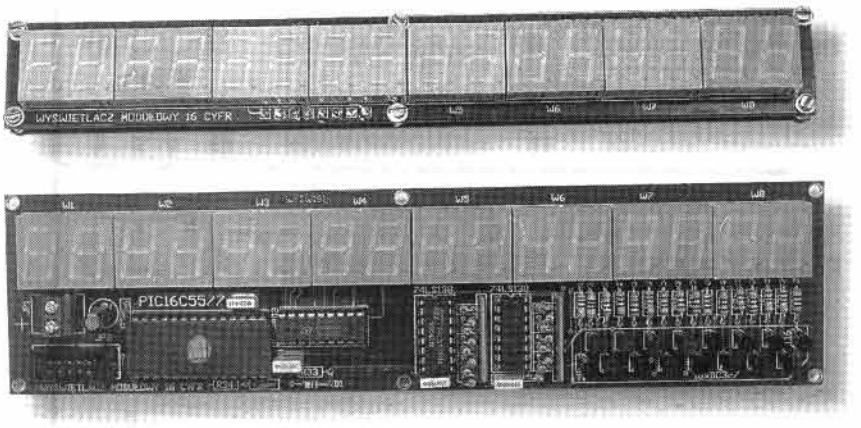
Modułowe wyświetlacze LED, część 1

kit AVT-159

Projektanci systemów mikroprocesorowych znają niedogodności związane ze sterowaniem grupą wyświetlaczy LED. Do tego celu jest bowiem zatrudniany procesor główny systemu, który w trybie przerwań co kilkaset mikrosekund, sekwencyjnie wyświetla znak po znaku. W przeciwieństwie do modułów LED, jakie łatwo jest sterować gotowym modułem LCD, który podczas obsługi wymaga przesłania tylko dwóch bajtów pod określony adres systemu.



W artykule przedstawimy wyświetlacz modułowy. Jednak nie będziemy robić konkurencji firmom wyspecjalizowanym w produkcji modułów LCD i moduł zaprojektowaliśmy w postaci zespołu wyświetlaczy siedmiosegmentowych LED.

Przewagą wyświetlaczy LED jest zdecydowanie większy kontrast zobrazowania, niż ma to miejsce we wskaźnikach LCD, korzystających ze światła odbitego. Lepsza czytelność wyświetlaczy LED jest okupiona zwiększonym poborem mocy, więc na pewno nasz moduł nie nadaje się do przenośnych urządzeń bateryjnych. Jednak do urządzeń stacjonarnych, zasilanych z sieci, nadaje się bardzo dobrze.

Przyjęto, że będzie to wyświetlacz o znacznej liczbie cyfr i małym „obciążeniu" programowym procesora systemu mikroprocesorowego. Wyświetlacz modułowy zajmuje jedną komórkę pamięci w systemie i wymaga dwóch linii kontroli współpracy z systemem. Zaprojektowano dwie wersje modułu różniące się liczbą cyfr. Wersja większa jest linijką 16 cyfr, zaś wersja uboższa zawiera 12 pozycji.

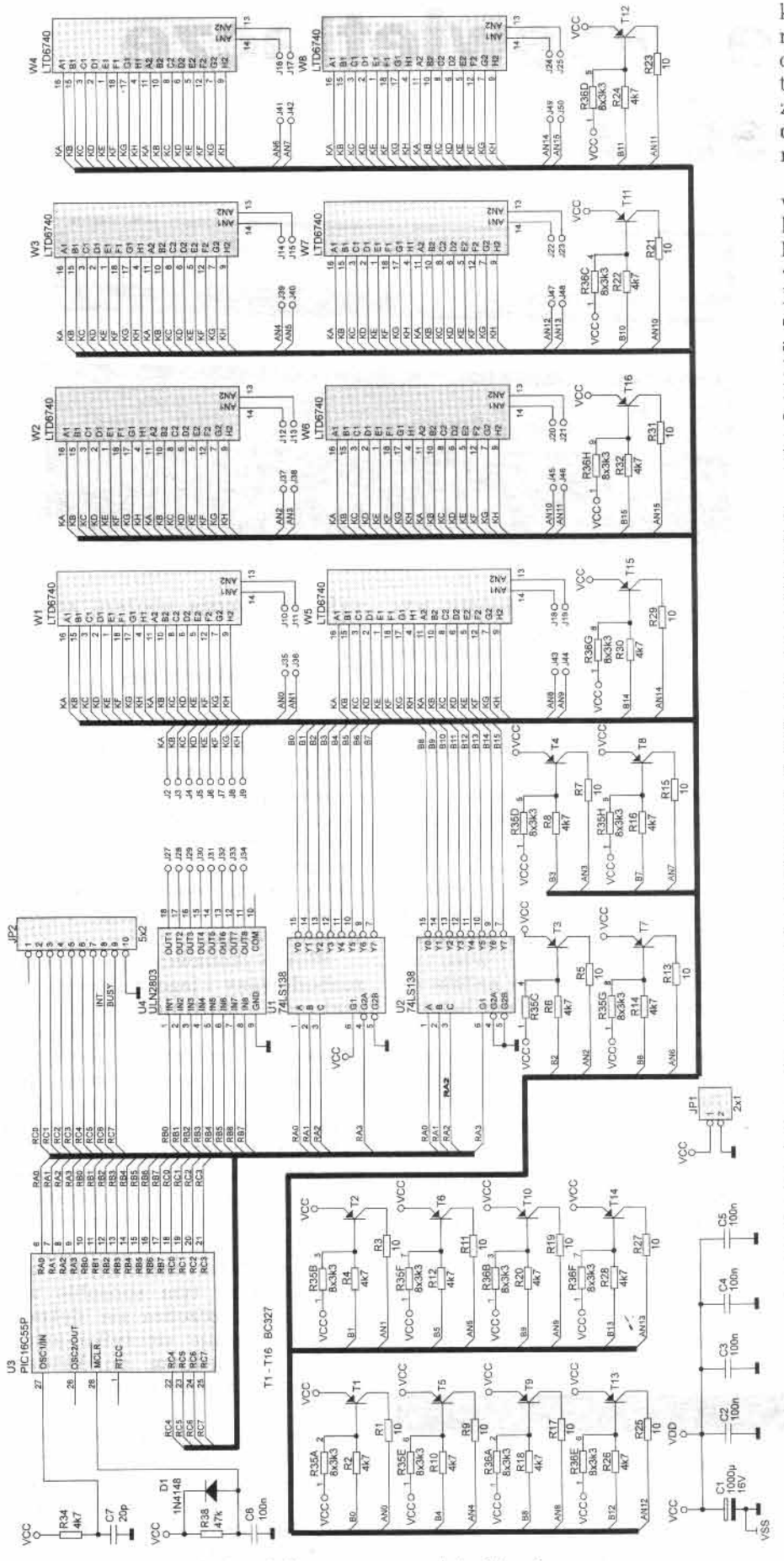
Parametry modułu	
Liczba cyfr	16 lub 12
Złącze komunikacyjne	6-bitowy interfejs równoległy z kontrolą transmisji
Napięcie zasilania	5V ±10%
Pobór prądu	zależny od jakości zastosowanych wyświetlaczy, np. dla HDSP 5321 nie większy od 200mA
Wymiary	
(wersja dwupłytkowa)	214x30x30mm (16cyfr) lub 163x30x30mm (12 cyfr)
(wersja jednopłytkowa)	214x54x20mm (16 cyfr) lub 163x54x20mm (12cyfr)

Układ elektryczny

Schemat układu elektrycznego modułu 16-cyfrowego przedstawiono na **rys.1**, natomiast wersji 12-cyfrowej na **rys.2**. Wersja 12-cyfrowa powstała z wersji 16-cyfrowej przez rezygnację z czterech cyfr z pozycji mniej znaczących (po prawej stronie). Moduł składa się z dwóch bloków funkcjonalnych: bloku sterującego i bloku wykonawczego.

Blok wykonawczy jest bardziej rozbudowany i jego zadaniem jest realizacja multipleksowego wyświetlania cyfr. Wyświetlanie multipleksowe polega na kolejnym i cyklicznym wyświetlaniu poszczególnych cyfr układu. Okres cyklu wyświetlania wszystkich cyfr („przemiatania") musi być wystarczająco krótki (sumaryczny czas przemiatania dla wszystkich wyświetlaczy nie powinien być dłuższy niż 10ms), aby nie widać było migotania świecących segmentów. Przy tym sposobie wyświetlania oszczędza się dekodery, gdyż stosuje się tylko jeden, zamiast po jednym do każdego wyświetlacza oraz energię zasilania.

Oczywiście, jasność świecenia segmentu wyświetlacza zależy od średniego prądu płynącego przez segment. Wskutek przełączania prąd ten będzie w przybliżeniu tyle razy mniejszy, ile jest w wyświetlaczu cyfr, dlatego aby uzys-



Rys. 1. Schemat układu elektrycznego modułu 16-cyfrowego.

kać tą samą jasność świecenia należy odpowiednio zwiększyć chwilową wartość prądu segmentów. Wykonuje się to przez zmniejszenia rezystancji ograniczającej prąd w obwodzie zasilania segmentów cyfr.

Użyte wyświetlacze mają wspólną anodę. W układzie modelowym były to wyświetlacze HDSP5321 firmy Hewlett-Packard, wytwarzające światło koloru malinowego. Naturalnie, mogą to być dowolne wyświetlacze ze wspólną anodą, o wysokości cyfr 13mm. Niewątpliwą zaletą wyświetlaczy HDSP5321 jest duża jasność świecenia, przy niewielkim średnim prądzie zasilania segmentu (rzędu 1mA).

Do sterowania anodami wykorzystano bardzo popularne tranzystory BC327, charakteryzujące się znacznym, bo równym 0.8A, maksymalnym prądem kolektora. Tranzystory te są zamykane w obudowie TO92, czyli właściwej dla tranzystorów małej mocy. Nie ma tu błędu, bowiem zauważmy, że tranzystory te będą w naszym układzie pracować jako klucze, czyli spadek napięcia na nich nie przekroczy 0.2V, a więc moc w nim wydzielana nie przekracza 160mW. W praktyce o wydzielaniu ciepła w tranzystorze decyduje moc średnia. Kluczujący tranzystor przewodzi tylko przez niewielki odcinek czasu, taka jest bowiem zasada działania wyświetlania multipleksowego. Można zatem nie obawiać się o ich przetrwanie w układzie.

Niski stan logiczny podawany na bazę jednego z tranzystorów T1-T16 włącza go, czyli na anodzie odpowiedniej cyfry pojawia się dodatnie napięcie, bliskie napięciu zasilania, pomniejszone jedynie o napięcie nasycenia przewodzącego tranzystora. Ów niski stan logiczny pochodzi z kolei z wyjść transkodera naturalnego kodu dwójkowego na kod 1z16, zbudowanego z dwóch demultiplekserów '138. Zaletą demultipleksera '138 jest niski wyjściowy stan aktywny, przez nas pożądany. Wybór właściwego demultipleksera jest zrealizowany przez odpowiednie podłączenie linii RA3 procesora do wejść zezwalających układów '138.

Powyżej został omówiono obwody wyboru cyfry, teraz należy omówić sterowanie katod segmentów, odpowiednio ze sobą połą-

czonych. Wzmacniaczem prądu segmentów jest układ ULN2803, zawierający osiem tranzystorów mocy, niezależnie sterowanych. Wysoki stan podany na wejścia układu ULN2803 włącza tranzystory, zamykając tym samym obwód zasilania odpowiednich segmentów w wybranej cyfrze. Segmenty poszczególnych cyfr są połączone równolegle, co wcale nie przeszkadza, bowiem zawsze jest polaryzowana tylko jedna anoda.

Część sterująca układem wykonawczym jest zbudowana na mikroprocesorze PIC16C55/57. Zadaniem mikroprocesora jest ustawianie właściwych sygnałów dla układu dekodera wyboru cyfry i wzmacniacza prądu segmentów. Kolejną funkcją mikrokontrolera jest współpraca z systemem nadrzędnym poprzez port C.

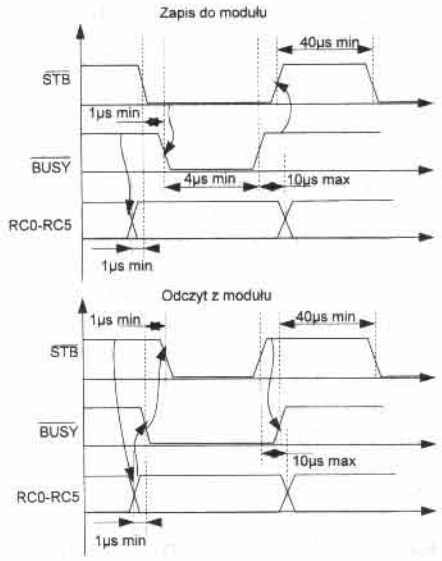
Komunikacja z modułem

Moduł współpracuje z systemem nadrzędnym poprzez złącze JP2. Przyjęto równoległy sposób transmisji z potwierdzeniem. Komunikacja z modułem jest dwukierunkowa: zapis do modułu i odczyt z modułu. Linie RC0-RC5 są liniami danych. W obu przypadkach transmisja zachodzi pod kontrolą dwóch linii portu C: RC6 i RC7.

Linia RC6, nazwana STB, w czasie zapisu do modułu służy do ustawienia niskiego stanu przez urządzenie nadające, którym informuje moduł o ustawieniu ważnych danych. Odpowiedzią modułu nadajnika jest niski stan na linii BUSY (linia RC7), który świadczy o potwierdzeniu początku transmisji. Zakończenie przyjmowania danych jest sygnalizowane zmianą BUSY z poziomu niskiego na wysoki, co oznacza, że dane zostały przez moduł odebrane. Dopiero wtedy na STB może pojawić się stan wysoki. Zależności czasowe pokazano na **rys. 3**.

W czasie odczytu danych z modułu linie kontroli współpracy zmieniają funkcje. W ten sposób kierunek przesyłania sygnałów kontrolnych jest zachowany i nie ma potrzeby przełączania statusu tych linii. W czasie odczytu danych z modułu linia BUSY pierwsza ustawia stan niski, potem następuje oczekiwanie na odpowiedź systemu nadrzędnego na linii STB. Stan wysoki na linii STB wymusza stan wysoki także na linii BUSY. Na rys. 3 pokazano również skrajne wartości czasów przełączenia sygnałów na liniach kontroli współpracy (STB i BUSY) oraz na liniach danych.

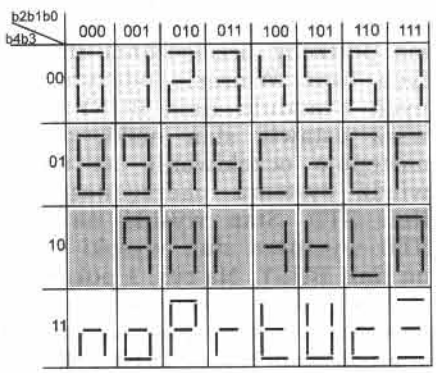
Rys. 2. Schemat układu elektrycznego modułu 12-cyfrowego.



Rys. 3. Zależności czasowe pomiędzy sygnałami STB, BUSY i liniami danych w czasie zapisu do modułu i odczytu z niego.

Procesor PIC przyjmuje sekwencje danych i odpowiednio je interpretuje. Przesyłane do modułu informacje mogą mieć dwojakie znaczenie. Mogą to być dane albo rozkazy. O ich znaczeniu decyduje bit b_5 odebranego słowa. Bit b_5 jest najstarszym bitem słowa i jego ustawienie oznacza przesyłanie rozkazu, zaś jego wyzerowanie danej. Wynika z tego, że moduł może rozpoznawać 32 różne znaki. Na **rys. 4** pokazano zawartość generatora znaków modułu. Pierwsze 16 znaków to cyfry kodu szesnastkowego, a pozostałe to litery, oczywiście o takim kształcie, jaki jest możliwy w oparciu o siedem segmentów. Układ kodowy generatora znaków pozwala transmitować dane liczbowe z systemu nadrzędnego bez dodatkowej konwersji.

Jak wspomniano, przesyłane dane to znaki i rozkazy dla modułu. Wprowadzono pojęcie kursora, czyli niewidocznego znacznika, określającego pozycję zapisu



Rys. 4. Zestaw znaków modułu.

Tabela 1

Kod rozkazu $b_5b_4b_3b_2b_1b_0$	Znaczenie
100001	Zapis nowej wartości do rejestru sterującego. Następnym odebranym przez moduł słowem powinna być nowa wartość słowa sterującego. Wykrycie kodu kolejnego rozkazu (b5=1) spowoduje interpretację odebranego słowa jako danej, z wymaskowaniem bitu b_5.
100010	Ustawienie kursora w nowej pozycji. Następnym odebranym przez moduł słowem powinna być nowa wartość pozycji kursora. Nawet przy b5=1 kolejne słowo będzie potraktowane jako dana, z wymaskowaniem bitu. Musi to być binarnie zapisana liczba od 0 do 15. Wartość 0 jest przypisana skrajnej lewej cyfrze, zaś wartość 15 (dla wersji 12-cyfrowej modułu jest to 11) dotyczy skrajnie prawej cyfry. Wartości powyżej 15 (11) będą przed użyciem traktowane jako modulo 16 (12).
100100	Odczytanie bieżącej pozycji kursora. Liczbą, która będzie mogła być odczytana będzie odpowiadająca pozycji kursora liczba od 0 do 15. Wartość 0 jest przypisana skrajnej lewej cyfrze, zaś wartość 15 (dla wersji 12-cyfrowej modułu będzie to 11) dotyczy cyfry skrajnej prawej.
101000	Odczytanie kodu znaku z bieżącej pozycji kursora. Liczbą, która będzie mogła być odczytana jest kodem znaku, jaki znajduje się na pozycji kursora. Kursor nie zmienia swego położenia.
110001	Wyczyszczenie zawartości wyświetlacza. Rozkaz ten nie wymaga dodatkowych cykli przesyłania danych. Na wszystkie pozycje modułu są wpisywane znaki spacji (kod 10H). Wyświetlacz jest wygaszony, łącznie z kropkami. Kursor ustawia się w skrajnym położeniu: na lewym skraju przy przesuwie cyfr w prawo (b_1b_0=00 w rejestrze kontrolnym) albo na prawym skraju modułu, gdy zadeklarowano przesuw cyfr w lewo (b_1b_0=01 w rejestrze kontrolnym). Jeśli kursor stoi (b_0=1 w rejestrze kontrolnym), to nie zmienia on swego położenia.
110010	Zaświecenie kropki. Następnym odebranym przez moduł słowem powinien być kodowany binarnie numer wyświetlacza, na którym ma być zapalona kropka. Wykrycie kodu kolejnego rozkazu będzie potraktowane jako dana, z wymaskowaniem bitu b_5. Wartość 0 jest przypisana skrajnej lewej cyfrze, zaś wartość 15 (w wersji uboższej jest to 11) dotyczy cyfry skrajnej prawej.
110100	Wygaszenie kropki. Następnym odebranym przez moduł słowem powinien być kodowany binarnie numer wyświetlacza, na którym znajduje się wygaszona kropka. Wykrycie kodu kolejnego rozkazu będzie potraktowane jako dana, z wymaskowaniem bitu b_5. Wartość 0 jest przypisana skrajnej lewej cyfrze, zaś wartość 15 (w wersji uboższej jest to 11) dotyczy cyfry skrajnej prawej.

znaku, który będzie dopiero odebrany z portu C. Sposób zapisu do wyświetlacza i ruch kursora jest zależny od trzech bitów rejestru sterującego, zdefiniowanego jako jedna z komórek pamięci danych procesora. Znaczenie bitów rejestru sterującego jest następujące:

b_0=0 - kursor przesuwa się po zapisaniu znaku na danej pozycji,
b_0=1 - kursor stoi, czyli znaki są zapisywane na jednej pozycji,
b_1=0 - przesuw kursora w prawo,
b_1=1 - przesuw kursora w lewo,
b_2=0 - kursor przesuwa się cyklicznie po wyświetlaczu,
b_2=1 - kursor dochodzi do brzegu wyświetlacza i tam się zatrzymuje.

Przy b_0=1 sposób ustawienia b_1 i b_2 nie ma oczywiście znaczenia.

Bit b_5 decyduje o tym, jak będzie interpretowana informacja przychodząca do modułu. Wyzerowany bit b_5 oznacza nadejście danej. Wykrycie odebrania danej spowoduje zapisanie kodu znaku do wyświetlacza (w miejscu położenia kursora), z jednoczesnym ustawieniem kursora na nowej pozycji, według ustawień słowa sterującego.

Ustawienie bitu b_5 oznacza rozkaz dla modułu. Zestaw rozkazów przedstawiono w **tabeli 1**.

Ustawianie kursora w nowej pozycji pozwala na podział cyfr modułu na grupy. Wtedy nie wszystkie cyfry mogą być wykorzystywane i niektóre z nich mogą być zawsze wygaszone, tworząc rodzaj separatora między grupami.

Odczyt bieżącej pozycji kursora zapewnia pełną kontrolę nad zapisem cyfr, co jest istotne w przypadku zapisu ciągów cyfr, pochodzących np. z klawiatury, czy przypadkiem nie są zapisywane poza wyświetlacz.

Odczyt kodu znaku z bieżącej pozycji kursora umożliwia traktowanie modułu jako rozszerzenia pamięci danych procesora głównego. Ma to niebagatelne znaczenie we współpracy z mikrokontrolerami, z natury rzeczy mającymi niewielką pamięć danych. Tym sposobem w procesorze głównym zwalniamy fragment pamięci, przeznaczony na bufor obrazu (wyświetlacza).

Wygaszanie całego modułu jest pomocne w sytuacjach zakończenia logicznego ciągu pewnych operacji i przejścia do realizacji kolejnego zadania systemu.

Z tabeli 1 i rys. 4 wynika, że kropka jest segmentem traktowanym w specyficzny sposób. Kropka dziesiętna służy przede wszystkim do oddzielenia części całkowitej od części ułamkowej liczby, zatem należało ją odseparować od wyświetlanych cyfr. Dodatkowo kropki mogą spełniać rolę prostych wskaźników stanu urządzenia, w którym moduł jest zamontowany.

Mirosław Lach, AVT