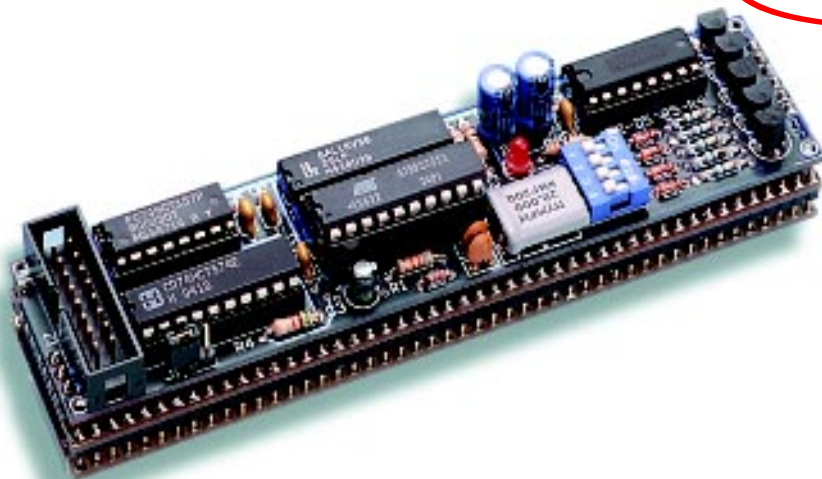


# Inteligentny wyświetlacz alfanumeryczny, część 1

## kit AVT-324

PROJEKT  
Z OKŁADKI



*Siedmiosegmentowe wyświetlacze LED umożliwiają od dawna najtańsze przedstawianie informacji w urządzeniach elektronicznych. Niestety, ze względu na ograniczone możliwości wyświetlania innych, poza cyframi, znaków alfabetu, nie nadają się do wielu zastosowań. Tam gdzie jest wymagane np. wyświetlenie tekstu coraz częściej stosuje się alfanumeryczne moduły LCD.*

*Co jednak zrobić, gdy informacja powinna być dobrze widoczna z większej odległości niż 1m? Problem ten rozwiązuje urządzenie opisane w artykule.*

Zamierzeniem autora było skonstruowanie inteligentnego, zgodnego z dotychczasowymi standardami, a przy tym uniwersalnego modułu wyświetlacza, który przy niewielkich rozmiarach i jednocześnie jak największym polu odczytowym można wykorzystać praktycznie wszędzie.

Najistotniejszymi zaletami przedstawionego rozwiązania są: możliwość kaskadowego łączenia maksymalnie 10 takich modułów, co zwiększa pojemność do 80 znaków oraz pełna kompatybilność ze światowym standardem interfejsu wyświetlaczy, wylansowanym przed kilkoma laty przez japoński koncern Hitachi.

Tak więc wszyscy użytkownicy znakowych wyświetlaczy LCD, opartych o standardowy sterownik typu HD44780, będą mogli w razie potrzeby zastosować nasze urządzenie, praktycznie bez przeróbek hardware'u i oprogramowania sterującego modułem.

Czytelnikom, którzy do tej pory nie zetknęli się z inteligentnymi wy-

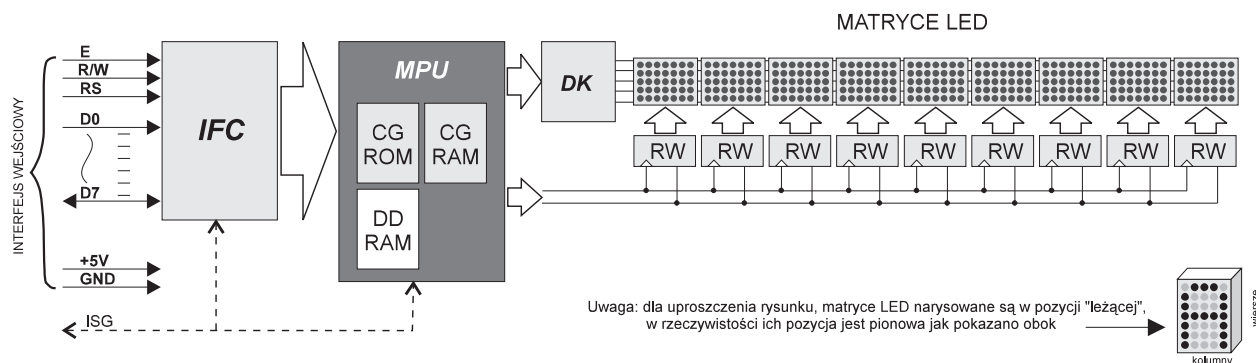
świetlaczami LCD, należy się wyjaśnienie, iż nasze urządzenie może spełniać rolę uniwersalnego wyświetlacza znakowego, sterowanego za pomocą dowolnego sterownika mikroprocesorowego, bądź opartego np. na pamięci EPROM. Dzięki temu, oraz dzięki możliwości kaskadowego łączenia wielu modułów jednocześnie, w prosty sposób będzie można budować np. tablice informacyjne, reklamowe, moduły wyświetlania w przyrządach pomiarowych, urządzeniach powszechnego użytku oraz wiele innych. W kolejnych numerach EP zaproponujemy wykonanie prostej przystawki opartej o pamięć EPROM/EEPROM, dzięki której nie wtajemniczony w technikę mikroprocesorową Czytelnik będzie miał możliwość sterowania przedstawionym urządzeniem.

### Opis układu

Schemat blokowy pojedynczego modułu przedstawia **rys.1**. Wiersze ośmiu matryc LED (5x7) są sterowane poprzez rejestry wierszy (RW), kolumny zaś za pośrednictwem wzmacniaczy 5 kolumn (DK). Sygnały dla RW i DK są generowane przez mikroprocesor - sterownik (MPU), który posiada wbudowany generator 112 znaków ASCII (CG ROM - ang.

#### Dane techniczne modułu AVT-324

- ✓ pole odczytowe: 8 znaków (matryce LED 5x7 punktów o wys. 18 mm),
- ✓ interfejs sterujący: kompatybilny z HD44780,
- ✓ sygnały sterujące: E, RS, RW, DO..D7 (tryby pracy: interfejs 4-bitowy lub 8-bitowy),
- ✓ monitorowanie zajętości dzięki fladze BUSY FLAG,
- ✓ pamięć wyświetlacza: maksymalnie 80 pozycji (10 modułów w trybie 2-liniowym),
- ✓ pamięć generatora znaków: 112 znaków ASCII, w tym wszystkie polskie znaki narodowe (duże i małe),
- ✓ możliwość programowania 8 znaków użytkownika,
- ✓ wbudowana obsługa kursora, atrybutu znaku i przesuwania tekstu,
- ✓ atrybuty znaku: normalny i migający,
- ✓ atrybuty kursora: normalny (2-wierszowy), pełny (cała matryca),
- ✓ dodatkowe instrukcje: czyszczenie pola, przesuwanie kursora, adresowanie pamięci wyświetlania (DD RAM) oraz generatora znaków (CG ROM i RAM),
- ✓ rozszerzenie: 2 linie po 5 modułów każda (80 znaków),
- ✓ napięcie zasilania: 5V,
- ✓ pobór prądu: typ. 280mA (standardowy tekst), maks. 500mA (zapalone wszystkie 280 pól),
- ✓ wymiary: 122,8 x 29,5 mm (4,83" x 1,16"), głębokość ok. 30 mm w zależności od sposobu montażu,
- ✓ konstrukcja umożliwiająca łączenie modułów w linie bez efektu zmiany rozstawu kolejnych pozycji wyświetlacza.



Rys. 1. Schemat blokowy prezentowanego modułu.

Character Generator ROM), pamięć znaków użytkownika (CG RAM) oraz pamięć przechowująca wyświetlany tekst (DD RAM - ang. Display Data RAM). Aby zapewnić kompatybilność sygnałów sterujących z pierwowzorem mikrosterownika LCD HD44780, niezbędne okazało się zastosowanie dodatkowego hardware'owego sprzęgu z interfejsem użytkownika (IFC). Dzięki temu, przyspieszono komunikację z modułem poprzez realizację obsługi flagi zajętości BUSY FLAG, implementowanej w sterownikach standardu Hitachi.

Do komunikacji ze światem zewnętrznym moduł wykorzystuje 11 sygnałów: 3 sterujące (E - enable signal, R/W - data read/write, RS - instruction/data select) oraz 8-bitową, dwukierunkową szynę danych (D0..D7). Dzięki nim użytkownik może wpisać znak lub rozkaz do modułu, lub odczytać jego stan. Znaczenie poszczególnych sygnałów oraz sterowanie modułem opiszemy w dalszej części artykułu.

Przy kaskadowym połączeniu wielu modułów sterowanych poprzez jedną szynę, niezbędne okazało się zastosowanie dodatkowego sygnału (ISG). Linia ta nie jest absolutnie związana z interfejsem użytkownika, a stanowi jedynie wewnętrzny sygnał do komunikowania się między poszczególnymi modułami rozbudowanego wyświetlacza.

Wszystkie sygnały sterujące oraz zasilanie wyprowadzono na dwurzędowe, standardowe złącze typu AFC-16. Rozkład wyprowadzeń przedstawia rys.2. Dodatkowy sygnał ISG jest na 16. wyprowadzeniu tego złącza. Dzięki takiemu rozwiązaniu, w prosty sposób, za pomocą jednej 16-żyłowej taśmy można „spinać” wiele mo-

dułów na raz, co czyni konstrukcję bardzo zwartą oraz ogranicza liczbę połączeń do minimum.

Schemat elektryczny części mikroprocesorowej modułu pokazany jest na rys.3.

Sercem układu jest układ U1. Zastosowano tani 8-bitowy mikrokomputer jednoukładowy, kompatybilny z procesorem 8051. Układ zawiera w swojej strukturze 128 bajtów wewnętrznej pamięci RAM, 2kB reprogramowalnej pamięci EEPROM typu „Flash” oraz szereg dodatkowych bloków funkcjonalnych, dzięki którym realizacja projektu została uproszczona do minimum. Mikroprocesor posiada dwa uniwersalne porty wejścia/wyjścia, dzięki którym wysyła wszystkie niezbędne sygnały potrzebne do sterowania matrycami LED i odbioru informacji wejściowej z interfejsu użytkownika. Znaczenie poszczególnych sygnałów na liniach portów jest następujące:

- DATA** - linia transmisji synchronicznej danych do 64-bitowego rejestru sterującego wiersze matryc LED (8 układów 74LS164),
- CLK** - sygnał taktujący (zegarowy) dla transmisji danych do rejestrów wierszy;
- D0/D1..D6/D7** - multiplexowane (z pomocą układu U4) linie wejściowe odebranego z interfejsu wejściowego 8-bitowego znaku lub rozkazu użytkownika;
- RS** - linia wejściowa odczytująca stan wejścia RS interfejsu ze złącza Z1;
- A/B** - sygnał wyboru bardziej lub mniej znaczącej „połówki” bajtu z linii D0..D7 złącza Z1;
- ACK** - sygnał wyjściowy zerujący flagę zajętości (BUSY FLAG), co informuje o zakończeniu wykonywania wewnętrznej operacji przez U1;

**A,B,C** - linie wyboru aktywnej (zapalanej) kolumny w matrycach LED;

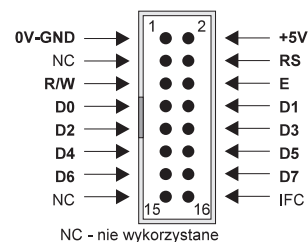
**INT** - sygnał informujący mikroprocesor U1, że w interfejsie wejściowym znajduje się gotowa do odbioru dana lub rozkaz;

**ADR** - wejście odczytu stanu przełączników adresowych SW1;

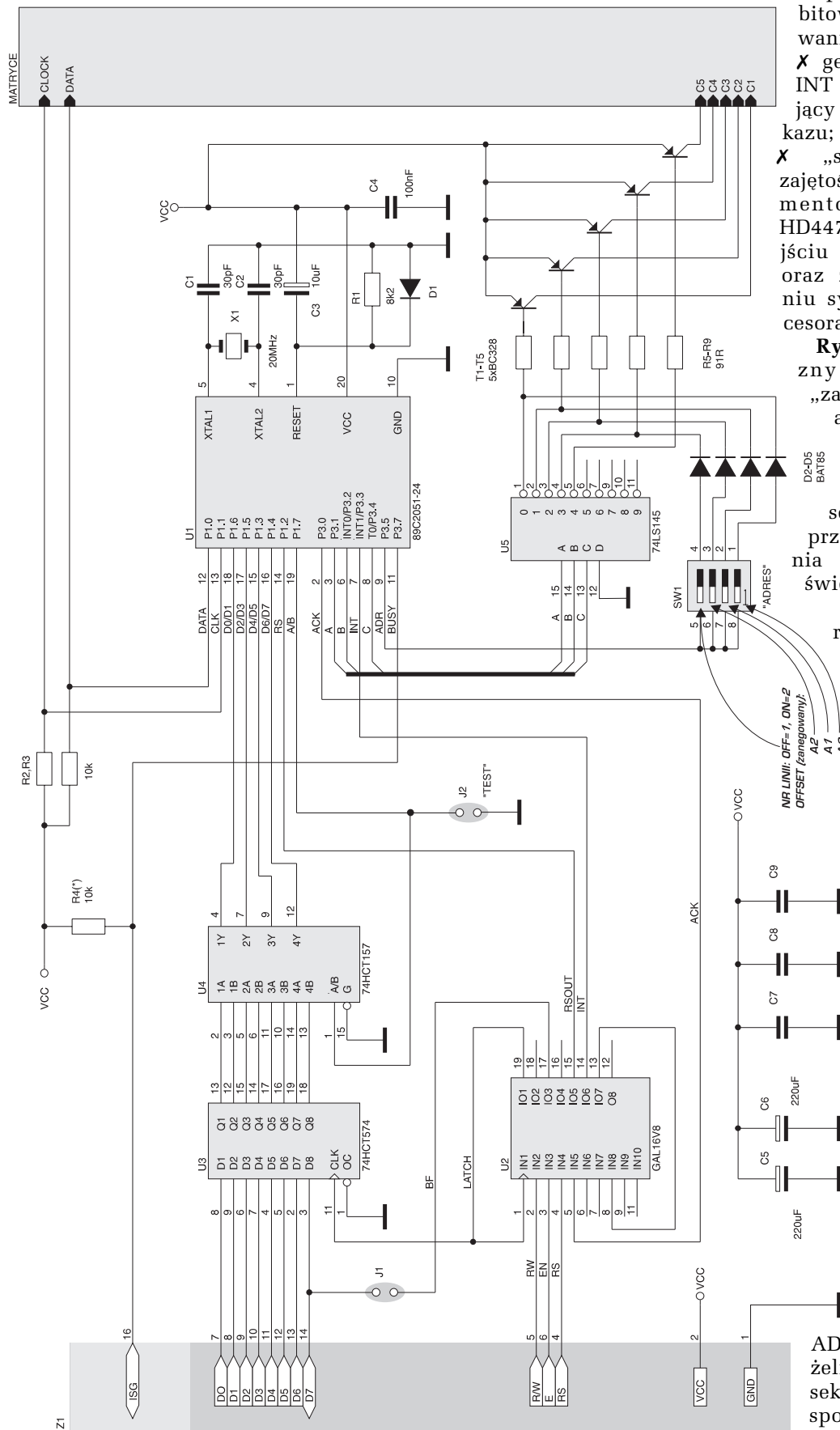
**BUSY** - sygnał zajętości modułu podczas wykonywanej operacji przez U1, wykorzystywany tylko do porozumiewania się pomiędzy połączonymi kilkoma modułami jednocześnie. Sygnał ten nazywany dalej „ISG” nie wymaga sterowania przez użytkownika.

Układ U3, wraz z programowaną matrycą logiczną typu GAL - U2, tworzą interfejs do komunikacji z systemem sterującym wyświetlaczem. Ze względu na niewystarczającą liczbę linii mikroprocesora U1 użyto dodatkowego układu 4-krotnego multipleksera 2-wejściowego U4. Jego zadaniem jest selekcja „połówek” bajtu z rejestru U3 i transmisja ich do mikroprocesora U1. W ten sposób za pomocą 5 linii (4 danych + 1 sterująca A/B) można przekazać 8-bitową liczbę do U1.

Układ U3 pełni rolę zatrasku danych pojawiających się na liniach D0..D7. Generacją sygnału strobowania tego układu zajmuje się układ U2. Oprócz tego, układ GAL pełni kilka dodatkowych funkcji, a mianowicie:



Rys. 2. Opis złącza wyświetlacza.



Rys. 3. Schemat elektryczny modułu sterującego.

X pełni rolę dodatkowego 1-bitowego rejestru do przechowania stanu wejścia RS (Z1);  
 X generuje sygnał przerwania INT dla procesora U1 informujący o nadejściu danej lub rozkazu;

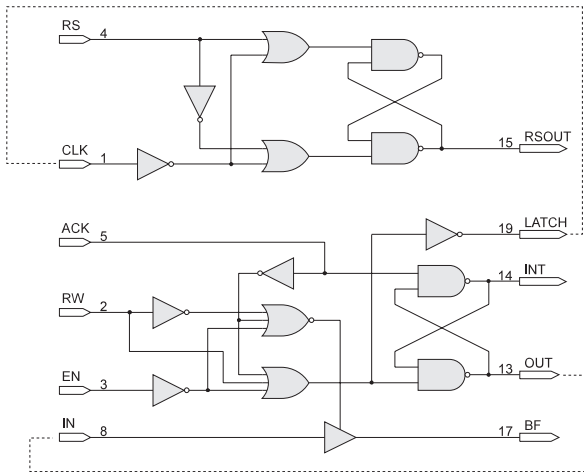
X „sprzętowo” ustawia flagę zajętości BUSY FLAG, implementowaną w sterownikach HD44780, natychmiast po nadejściu danych na linii D0..D7 oraz zeruje ją po potwierdzeniu sygnałem ACK z mikroprocesora U1.

Rys.4 przedstawia wewnętrzny układ kombinacyjny „zaszyty” w strukturze GAL-a U2. Dzięki takiemu rozwiązaniu uniknięto użycia kilku dodatkowych, standardowych układów serii TTL LS lub HCT, co przyczyniło się do zmniejszenia gabarytów modułu wyświetlacza.

Liniami przerywanymi na rysunku zaznaczono zewnętrzne sprzężenia, niezbędne do prawidłowego funkcjonowania układu U2.

Do sterowania tranzystorami załączającymi kolumny w matrycach LED (anody diod) służy układ U5 - dekodery kodu BCD na 1 z 10, z wyjściami typu OC. Mikroprocesor U1 podając odpowiednią kombinację sygnałów A,B,C na wejścia 15,14,13 dekodera U5, powoduje pojawienie się stanu „0” na jednym z jego wyjść i w konsekwencji załączenie jednego z pięciu tranzystorów T1..T5.

Do wyjść dekodera dołączono także, poprzez diody D2..D5, przełącznik 4-sekcyjny SW1. Pojawiające się logiczne „0” na jednej z katod diod powoduje zwarcie do masy wejścia ADR mikroprocesora U1, jeżeli zwarta jest odpowiednia sekcja SW1. W ten prosty sposób, za pomocą tylko jednej linii ADR, procesor odczytuje stan całego przełącznika. Rezystory R2 i R3 podciągają



Uwaga: linią przerywaną zaznaczono połączenia na zewnątrz układu GAL (na pcb)

Rys. 4. Schemat przedstawiający wnętrze GALa.

linie DATA i CLK procesora U1 do plusa zasilania. Jest to konieczne ze względu na to, że końcówki P1.0 i P1.1 kontrolera U1, w trybie pracy jako wyjścia, są typu otwarty dren. Dodatkowy rezystor R4 spełnia podobną rolę dla linii BUSY procesora U1. Sposób montażu tego elementu oraz znaczenie zwory J1 przedstawimy w dalszej części artykułu podczas opisu uruchamiania modułu.

Elementy X1, C1 i C2 stanowią zewnętrzny obwód generatora zegarowego procesora U1. Zastosowanie kwarcu o częstotliwości podstawowej 20MHz przyspieszyło pracę całego urządzenia oraz jego reakcje na sterowanie z zewnątrz.

Rezystor R1, dioda D3 i kondensator C3 umożliwiają prawidłowy start procesora U1 po włączeniu napięcia zasilającego.

Dodatkowe kondensatory C4, C7..C9 blokują linie zasilające w pobliżu układów scalonych na płytce sterownika, a C5 i C6 dodatkowo filtrują napięcie zasilające.

Na rys.5 jest przedstawiona część układu związana ze sterowaniem wierszy poszczególnych matryc. Jak widać, dzięki zastosowaniu 8-bitowych rejestrów 74LS164 liczbę linii sterujących ograniczono do niezbędnego minimum - 7 linii.

Rejestry U6..U13 pełnią także rolę wzmacniaczy mocy, bowiem bezpośrednio sterują one katodami wszystkich 8 matryc LED. Rezystory R11..R66 ograniczają prąd wpływający do wyjść rejestrów do wartości bezpiecznej. Zbyt uproszczone, wydawać by się mogło,

sterowanie matrycami LED w praktyce okazało się wystarczające. Przy zastosowaniu markowych matryc, np. Kingbright typ TA07-11, jasność wyświetlanej informacji jest wysoka, a układy U13..U20 nie nagrzewają się powyżej 40°C podczas wielogodzinnej pracy całego modułu.

Rejestry wierszy połączone są szeregowo, a ostatecznie wyjście bajtu danych QH połączone jest z wejściem następnego rejestru. Wszystkie kostki U6..U13 taktowane są oczywiście wspólnym sygnałem zegarowym. Wejścia zerujące CLR zostały niepodłączone, bowiem czyszczenie wyświetlacza - rejestrów 74LS164, odbywa się poprzez wpisanie samych jedynek przez mikroprocesor U1 za pośrednictwem linii DATA.

Ostatnie, najstarsze wyjście QH układu U13 steruje dodatkową diodą LED, która fizycznie umieszczona jest na tylnej ściance modułu. Zadaniem tego elementu jest wizualne informowanie operatora (np. serwisanta) o poprawnej pracy urządzenia. Dioda D6 świeci, gdy moduł jest w trakcie odbierania i przetwarzania informacji wprowadzanej na złącze Z1. Gaśnie, gdy moduł jest wolny i oczekuje na przyjęcie następnego polecenia. Oczywiście, w praktyce nie musimy montować tej diody, lecz jej użycie jest pomocne, szczególnie gdy budujemy wyświetlacz złożony z większej liczby modułów AVT-324.

Kondensatory C10..C17 blokują linie zasilające układy rejestrów 74LS164.

Na schemacie z rys. 5 przedstawiono dodatkowo wygląd zastosowanej matrycy LED oraz jej strukturę wewnętrzną. Z układu połączeń 35 diod LED w matrycy łat-

wo jest wywnioskować sposób jej sterowania w przedstawionym układzie wyświetlacza.

### Zasada działania

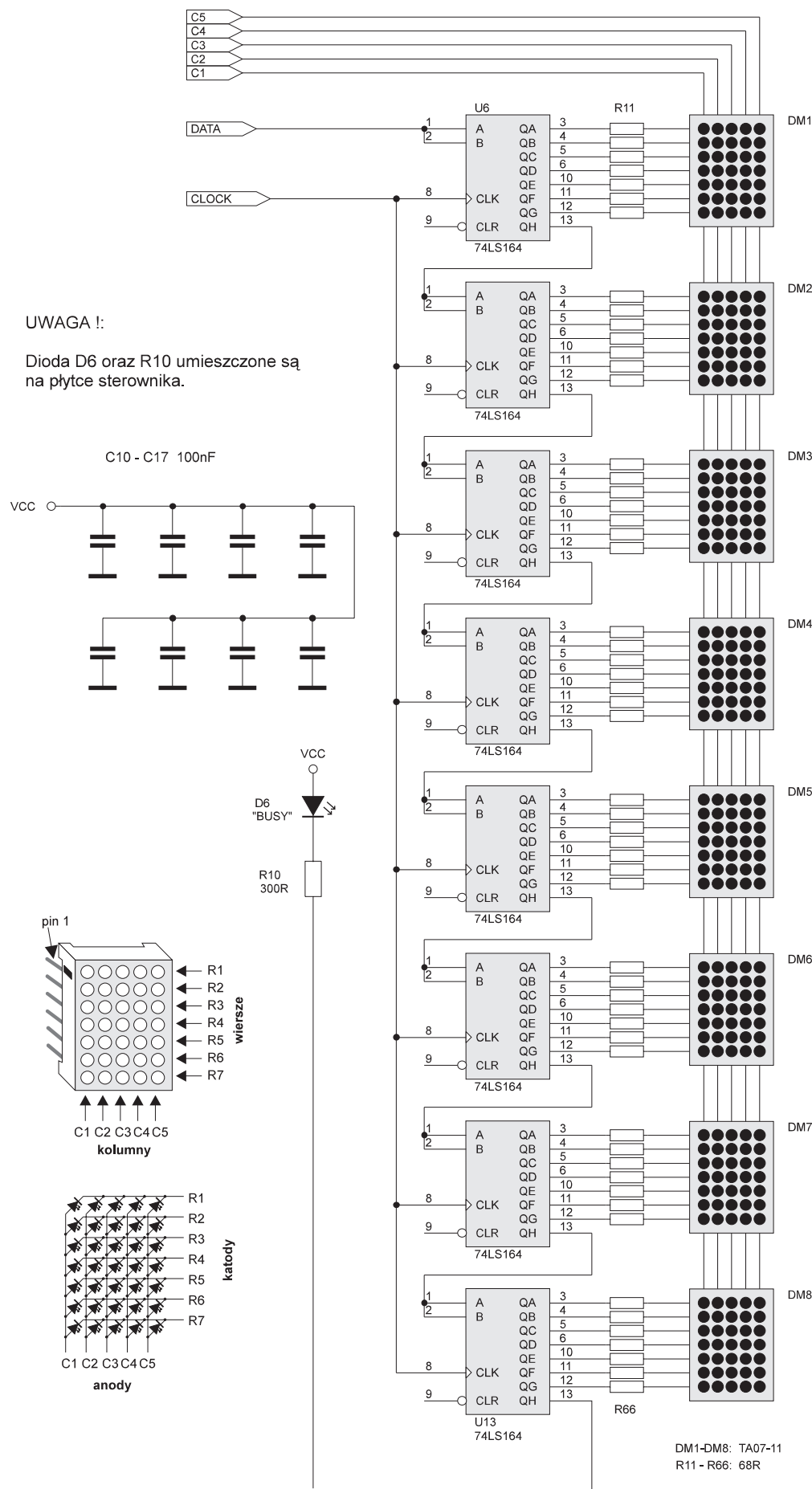
Po zapoznaniu się ze wszystkimi blokami funkcjonalnymi urządzenia możemy przeanalizować sposób jego działania. Na początku wyjaśnimy znaczenie poszczególnych sygnałów sterujących modułem (tabela 1).

W tabeli nie ujęto dodatkowo sygnału ISG, który nie jest związany z interfejsem użytkownika, a służy jedynie do komunikowania się pomiędzy pojedynczymi wyświetlaczami AVT-324 w wypadku pracy wielomodułowej.

Przeanalizujemy sytuację, kiedy zewnętrzny układ sterujący pracą modułu zapisuje kolejny znak do wyświetlenia. Rys.6a przedstawia pojedynczy cykl zapisu do interfejsu wejściowego Z1. Po odpowiednim ustawieniu sygnałów sterujących: R/W=0 bo zapis, RS=1 bo dana do zapisu, a następnie podaniu na szynę danych D0..D7 bajtu do zapisu, odblokowujemy wyświetlacz podając dodatni impuls na wejście E (en-enable). Patrząc na rys.4 zauważmy, że wraz z nadejściem narastającego zbocza sygnału E ustawione zostaną przetrzutniki /RS w układzie GAL. Poczynając od góry (rys.4): na wyjściu przetrzutnika RSOUT ustawi się stan logiczny jaki jest na pinie

Tabela 1.

Pin na Z1	Symbol	Poziom	Znaczenie
1	GND	-	masa zasilania
2	Vcc	-	zasilanie +5V
3	NC	-	nie dołączone
4	RS	H/L	"data/instruction select" - - rodzaj informacji na wejściu D0..D7 "H" gdy instrukcja "L" gdy dana (znak do wyświetlenia)
5	R/W	H/L	"read/write" - - sygnał odczytu lub zapisu do modułu "H" gdy odczyt "L" gdy zapis
6	E (EN)	L→H→L	"enable signal" - uaktywnienie wyświetlacza
7	D0	H/L	linie danych D0..D7
8	D1	H/L	
9	D2	H/L	
10	D3	H/L	
11	D4	H/L	
12	D5	H/L	
13	D6	H/L	
14	D7	H/L	



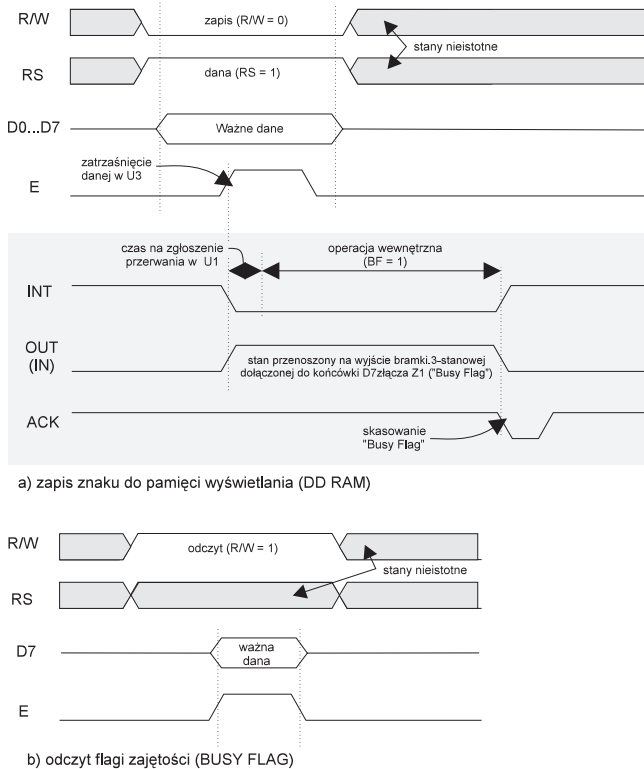
Rys. 5. Schemat elektryczny płytki wyświetlaczy.

wejściowym RS, wyjście drugiego przerzutnika INT przejdzie w stan niski generując sygnał przerwania na wejściu procesora U1 (INT1 - patrz Rys.3). Dzięki zewnętrznemu połączeniu wyjścia GAL oznaczonego jako OUT z wejściem IN uzyskamy możliwość monitorowania (odczytu) wyjścia BF, które zostaje odblokowane sygnałem zezwolenia bramki 3-stanowej. Dokładnie sytuację tę przedstawia Rys.6b. Zastosowanie trójstanowej bramki w strukturze GAL pozwala na włączenie modułu (lub wielu modułów) w dowolny system mikroprocesorowy, bez potrzeby stosowania dodatkowych buforów pośredniczących.

Ze względu na aktywowanie interfejsu IFC sygnałem E, a właściwie jego narastającym zboczem, parametry czasowe przebiegów z rys.6a i b nie są krytyczne, a wartości minimalne opóźnień zawierają się w granicach propagacji dwóch połączonych szeregowo bramek typu HCT, a więc kilkunastu nanosekund.

Także czas trwania sygnału E może wynosić poniżej 100 ns, z tego względu, że steruje on przerzutnikiem typu /RS (patrz rys.3). Istotne jest aby przed nadejściem tego sygnału ustawić linie R/W, RS oraz dane na szynie D0..D7. Resztą zajmuje się układ IFC i mikroprocesor.

Wyjaśnienia wymaga jeszcze sposób komunikacji pomiędzy wieloma modułami. Otóż, aby zapewnić możliwość łączenia kilku modułów w jeden większy wyświetlacz, za pomocą tej samej magistrali sterującej (E,R/W, RS, D0..D7), należało uporządkować informację, która napływa do każdego modułu w tej samej chwili. Najlepszym rozwiązaniem tego problemu okazało się „a-



Rys. 6. Przebiegi charakteryzujące interfejs wyświetlacza.

dresowanie“ pola odczytowego każdego modułu.

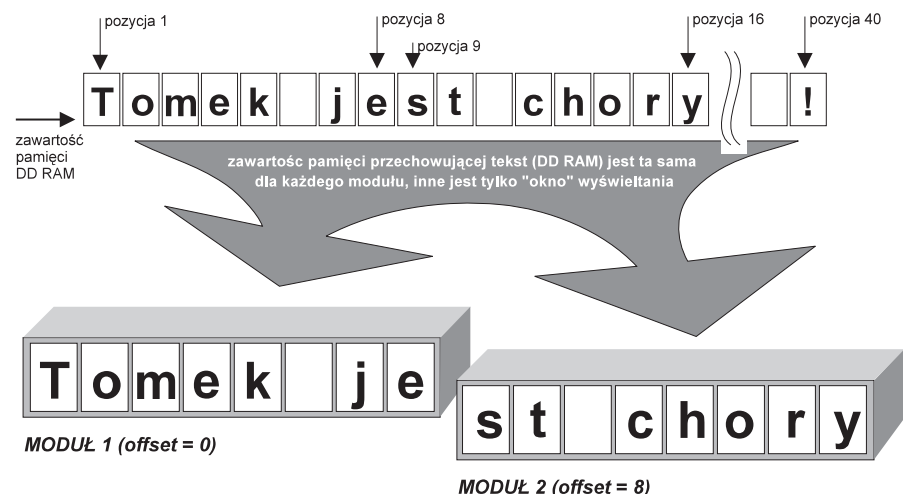
Sytuację tę dokładnie wyjaśnia rys.7. Każdy moduł fizycznie jest takim samym urządzeniem, zawiera 40-pozycyjną pamięć do przechowywania wyświetlanych znaków (a dlaczego nie 80-znakową - wyjaśnimy za chwilę), taki sam generator 112 znaków ASCII oraz taką samą pamięć generatora znaków użytkownika. Różnica jedynie polega na określeniu offsetu, czyli przesunięcia tzw. okna wyświetlania dla każdego modułu oddzielnie. Zauważmy wszakże, że moduł posiada jedynie 8 pozycji do wyświetlenia znaku, a wewnętrzna pamięć przechowuje ich aż 40. Dzięki takiemu rozwiązaniu sterowanie kilkoma modułami „spiętymi“ w jeden większy wyświetlacz jest takie same jak pojedynczym modułem.

Należy jedynie, po fizycznym zamocowaniu poszczególnych modułów w urządzeniu docelowym, ustawić odpowiedni adres wspomnianego „okna“. W praktyce adres ustawiany jest za pomocą umieszczonych na płycie każdego modułu trzech mikroprzełączników, o których wspomnieliśmy wcześniej (SW1).

Te same uwagi odnoszą się do sterowania kursorem, który jest

ny, czy zgaszony.

Ze względu na ograniczony rozmiar pamięci RAM w mikroprocesorze sterującym modułem U1, obszar adresowy podzielono na dwie linie po 40 pozycji każda. Fizycznie numer linii jest dodatkowym adresem każdego modułu a wyboru jej dokonuje się za pomocą czwartego mikroprzełącznika. Różnica między opisanym wcześniej offsetem okna polega jednak na tym, że każda linia wyświetlacza (złożona maksymalnie z 5 modułów) zawiera odrębną pamięć wyświetlanych znaków.



Rys. 7. Sposób wyświetlania informacji na kilku połączonych modułach.

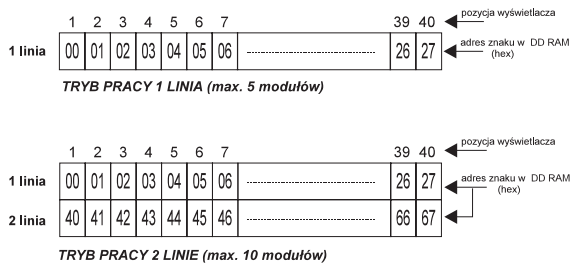
dostępny w naszym module. Przy pracy wielomodułowej kursor znajduje się zawsze tylko w jednym miejscu na określonej pozycji wyświetlacza. Przesuwanie kursora np. z ostatniej matrycy modułu nr 1 na pierwszą pozycję modułu nr 2 następuje automatycznie, tak jak w przypadku obsługi wyświetlanego tekstu. Istotne jest, że kursor zawsze wskazuje na aktualny adres w pamięci DD RAM, niezależnie od tego czy jest zapalony

Pierwsza linia wyświetlacza mieści pamięć wyświetlania (DD RAM) o adresach 00h..27h (40 pozycji), druga zaś to adresy: 40h..67h (także 40 pozycji), co w sumie pozwala na pokazanie, wspomnianych w danych technicznych, 80 znaków.

W przypadku, gdy użytkownik wysłał znak pod adres np. 55h, moduły z ustawioną linią 2 zaakceptują go i umieszczą w swojej pamięci DD RAM. W efekcie ten z modułów w linii 2, którego offset jest równy 16 (10 hex), wyświetli na 6-tej (5-tej numerując od 0) matrycy LED żądany znak. Obliczmy: 40h + 10h + 5 = 55h = adres, 40h - bo 2 linia, 10h - offset drugiego modułu, 5 - numer matrycy LED. Oczywiście, moduły z linii 1 zignorują ten odebrany znak.

Użytkownicy wyświetlaczy tekstowych LCD zauważą w tym miejscu logiczną konsekwencję takiej organizacji pamięci wyświetlania znaków (DD RAM), która przy niedostatecznej ilości pamięci RAM pozwala na zachowanie kompatybilności programowej z punktu widzenia użytkownika. Rys.8 ilustruje organizację pamięci przy wyborze sterowania wyświetlaczem 1 i 2-liniowym. Jak widać, w porównaniu z pierwowzorem (sterownikiem LCD HD44780) nasze rozwiązanie posiada ograniczenie maksymalnej liczby wyświetlanych znaków w trybie 1-liniowym tj. do 40 pozycji.

Jednak przy pracy 2 linie po 40 znaków wyświetlacz zachowuje się jak przewiduje wspomniany standard sterowników LCD.



Rys. 8. Adresowanie poszczególnych pozycji wyświetlacza.

Dodatkowy sygnał ISG (rys.1), służący do komunikowania się modułów w obrębie jednego wyświetlacza, spełnia właściwie rolę wspólnej linii do analizy zajętości każdego z modułów.

Otóż zgodnie z przebiegiem z rys.6a po zapisie danej (RS=1) lub rozkazu (RS=0) do modułu, mikroprocesor potrzebuje określonego czasu na wykonanie wewnętrznych operacji, polegających na odebraniu danej z interfejsu IFC oraz umieszczeniu jej na wyświetlaczu. Czas ten z reguły nie przekracza kilkudziesięciu mikrosekund, a najlepszą wspomnianą już metodą jest odczytywanie flagi zajętości (BUSY FLAG).

Tak więc, w momencie zgłoszenia przerwania (INT) przez układ GAL - U2, mikroprocesor U1 najpierw ustawia stan niski na dodatkowej linii ISG (patrz rys.2). Następnie odczytuje daną z rejestru U3, itd. Kiedy wszystkie operacje zostaną zakończone, układ U1 zaczyna badać stan linii ISG, do której przecież są dołączone pozostałe moduły. Mikroprocesor każdego modułu czeka na moment, kiedy na linii wystąpi ponownie stan wysoki. Ponieważ podczas odczytu przez mikroprocesor stanu końcówki P3.7, pin portu znajduje się w stanie wysokiej impedancji, niezbędne stało się zastosowanie dodatkowego rezystora podciągającego R4. Rezystor taki może być wlutowany tylko w jednym ze wszystkich połączonych modułów. Aczkolwiek wyposażenie w niego każdego modułu AVT-324 nie wpłynie ujemnie na pracę całego układu.

Zauważmy teraz, że jeżeli mikroprocesor w każdym module zakończy operację wewnętrzną, to na linii ISG zgodnie z zasadą iloczynu montażowego powstanie stan wysoki. Procesor U1 wykrywając to, wyśle sygnał ACK do układu GAL - U2, który skasuje

ustawioną wcześniej flagę zajętości BUSY FLAG. Na wyjściu BF układu U2 ustali się (przy odczycie) stan niski, co sygnalizuje fakt gotowości do przyjęcia kolejnego znaku lub rozkazu przez wyświetlacz. Ze względu na fakt wystąpienia ta-

kiego stanu w każdym z modułów zastosowano zworę J1. Zworę tę należy montować tylko w jednym module całego wyświetlacza, pozostawiając w pozostałych sygnał z końcówki 17 U2 odcięty.

Wnikliwy czytelnik zauważy, że taki sposób arbitrażu na linii ISG jest prosty, a jednocześnie poprawny. Pomimo że w każdym module moment zakończenia wykonywania wewnętrznych operacji może być nieco różny w czasie, to zawsze cały wyświetlacz jest gotowy do odebrania następnej informacji w jednej chwili (kiedy ostatni z modułów będzie gotów).

Na koniec istotna uwaga dotycząca dodatkowej funkcji, jaką spełnia sygnał INT pochodzący z układu U2. Otóż, oprócz roli zgłoszenia przerwania dla procesora U1, synchronizuje on także wewnętrzne zegary wszystkich modułów w przypadku pracy wielomodułowej. Wykorzystano tu fakt jednoczesnego nadejścia danej do interfejsu IFC każdego z modułów. Drobne różnice (rzędu mikrosekund) w synchronizacji migania przy częstotliwości 2Hz nie mają wpływu na efekt wyświetlania migającego kursora lub znaków z ustawionym atrybutem.

Synchronizacja taka okazała się niezbędna, bowiem przy dłuższej pracy kilku połączonych modułów lub przy nierównomiernym ich starcie po włączeniu zasilania, występował efekt „rozjeżdżania” się efektu migotania znaków i symboli.

**Sławomir Surowiński, AVT**

*Opracowanie oprogramowania sterującego przedstawionym urządzeniem było wspomagane „Emulatorem procesora 87C51” który jest dostępny jako kit AVT-288.*

*Układy U1 w wersji handlowej programowano „Programatorem procesorów MCS-51” - kit AVT-320.*

## WYKAZ ELEMENTÓW

### Rezystory

R1: 8,2kΩ  
R2..R4: 10kΩ  
R5..R9: 91Ω  
R10: 300Ω  
R11..R66: 68Ω

### Kondensatory

C1, C2: 27..30pF  
C3: 10μF/16V  
C4, C7..C17: 100nF  
C5, C6: 220μF/6,3V

### Półprzewodniki

U1: 89C2051-24MHz zaprogramowany AVT-324  
U2: GAL16V8B zaprogramowany AVT-324  
U3: 74HCT574  
U4: 74HCT157  
U5: 74LS145  
U6..U13: 74LS164 (nie stosować zamienników)  
T1..T5: BC327..9  
D1: 1N4148 lub odpowiednik  
D2..D5: BAT85 (BAT43, lub inna małej mocy Schottky’ego)  
D6: LED 3mm  
DM1..DM8 ? : TA07-11 (EWA,GWA lub YWA) Kingbright lub odpowiednik

### Różne

X1: rezonator kwarcowy 20MHz  
SW1: SW-DIP 4  
J1: goldpin 1x2 + jumper  
Z1: gniazdo AWP-16 proste wtyk AFC-16, 1szt.  
taśma 16-żył, 30 cm  
podstawki pod układy: U1..U5

**Uwaga:** w skład kitu AVT-324B nie wchodzi matryce LED. Można je zamawiać oddzielnie w Dziale Handlowym.