

Timer mikroprocesorowy



Opisywany układ przeznaczony jest do kontroli czasu wystąpienia mówcy podczas wykładu czy przemówienia. Odlicza czas: minuty i sekundy. Maksymalna pojemność wynosi 99 minut 59 sekund. Posiada także inne praktyczne i przydatne funkcje.

W artykule opisane są dwie wersje timera. Jedna z bardzo dużym wyświetlaczem (o wysokości cyfr 57 i 38mm), druga z typowymi wyświetlaczami o wysokości cyfr 20 i 14mm. Pokazany na fotografiach prototyp z wielkimi wyświetlaczami i dodatkowym timerem kontrolnym umieszczonym w obudowie zasilacza/sterownika, pracuje w Warszawie w sali wykładowej przy ul. Szwedzkiej.

Opis układu

Rysunek 1 ilustruje zasadę działania systemu kontroli czasu mówcy. Timer, a właściwie dwa timery, zaczynają liczyć czas od zera w górę po włączeniu napięcia zasilającego. Mogą być w każdej chwili wyzerowane przez wyłączenie zasilania na co najmniej sekundę. Dodatkowo, jeśli w czasie pracy obsługa zmieni biegunowość napięcia zasilającego, timery będą nadal liczyć czas, ale wskaźnik będzie migał, wskazując mówcy, że już przekroczył przewidziany czas i powinien szybko skończyć wystąpienie.

Przewidziano także możliwość lokalnego zerowania głównego timera za pomocą wbudowanego weń przycisku.

Rysunek 2 pokazuje schemat ideowy wersji z mniejszymi wyświetlaczami. Podstawą konstrukcji jest popularny „atmelek” 89C2051. Mostek prostowniczy D1...D4 jest niezbędny, by układ pracował przy dowolnej biegunowości napięcia zasilającego.

Uwaga! Mimo obecności mostka prostowniczego, układ nie jest przystosowany do zasilania napięciem zmiennym. Wskazuje na to między innymi niewielka wartość kondensatora filtrującego C1. Za to kondensator umieszczony za stabilizatorem ma wyjątkowo dużą wartość 470μF. Nietypowym elementem jest także rezystor R1, włączony „bezproduktywnie” między szyny zasilania. Jest on niezbędny, by rozładować C3 do zera, a tym samym zapewnić niezawodne zerowanie nawet po krótkotrwałym wyłączeniu napięcia zasilania. Drugim sposobem wyzerowania licznika jest naciśnięcie przycisku S1. Program wykryje stan niski na tej nóżce i wyzeruje stosowne rejestry.

Cztery cyfry wyświetlacza pracują oczywiście w trybie multipleksowym. Wykorzy-

stano wskaźniki LED ze wspólną anodą. Ze względu na małą wydajność portów w stanie wysokim, tranzystory T1...T4 są konieczne, byysterować znacznym prądem anody wyświetlaczy. Ponieważ prąd anod jest dość duży, zamiast popularnych BC558 zastosowano małe „darlingtony” BC516 o większym prądzie dopuszczalnym. Wydajność prądowa portu P1 w stanie niskim jest dość duża i wynosi 20mA, co jednak przy takim sposobie sterowania niestety nie wystarcza. Dlatego dodano tranzystory T5...T11. I tu zamiast popularnych BC558 wykorzystano BC327 o większym prądzie kolektora. Rezystory R8...R14 decydują o jasności świecenia wskaźników.

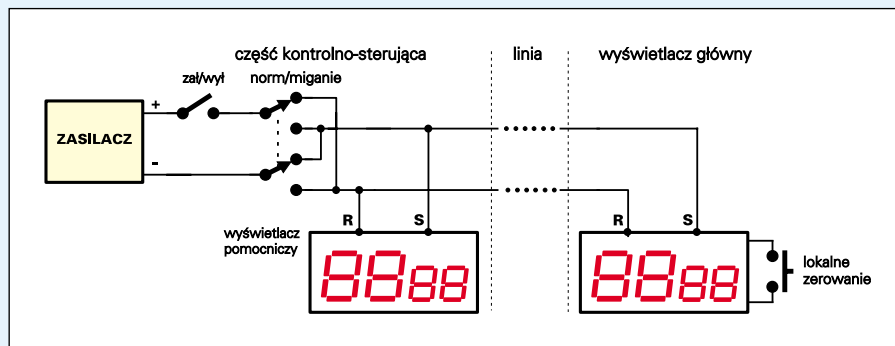
W wersji podstawowej diody D7...D10 nie będą montowane – zamiast nich należy wlutować zwory.

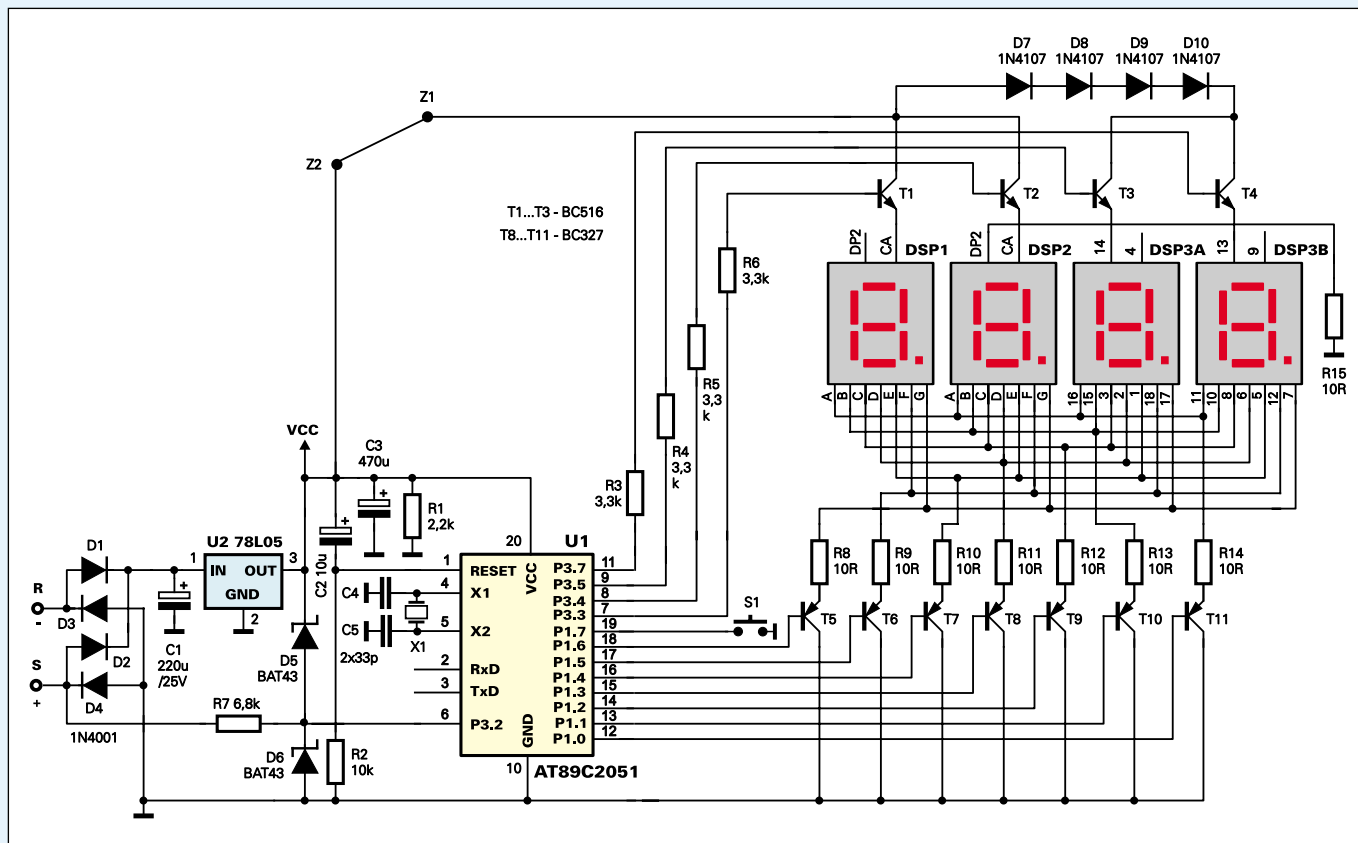
Elementy R7, D6, D6 pracują w obwodzie kontroli biegunowości napięcia zasilającego. W czasie normalnej pracy na punkt S trzeba podać dodatni biegun napięcia zasilania. Wtedy na końcówce P3.2 (nóżka 6) występuje stan wysoki, a timer normalnie zlicza i wyświetla czas.

Gdy biegunowość napięcia podawanego na punkty R, S zmieni się, na końcówce P3.2 pojawi się stan niski. Program to wykryje i wyświetlacz będzie migać. Dodatkowo, w czasie migania jasność segmentów jest dwukrotnie większa, przez co dodatkowo zwraca uwagę mówcy. Zwiększanie jasności zrealizowano w sposób programowy, bo takie rozwiązanie okazało się najprostsze.

Wersję z dużym wyświetlaczem, pokazaną na fotografii wstępnej zrealizowano według nieco innego schematu, pokazanego na **rysunku 3**. Jeden segment dużego wyświetlacza zawiera kilka diod LED połączonych w szereg, co oczywiście wymaga zastosowania napięcia zasilającego znacznie wyższego

Rys. 1





Rys. 2 Schemat ideowy wersji „małej”

niż 5V. Próby wykazały, że napięcie zasilające musi wynosić około 15V. Ponieważ mikroprocesor musi nadal być zasilany napięciem około 5V, konieczne jest dodanie czterech tranzystorów pośredniczących (T1...T4). Konieczne było też dodanie czterech diod D7...D10, by wyrównać jasność świecenia wszystkich wyświetlaczy. Rzecz w tym, że zastosowane mniejsze wskaźniki sekund mają segmenty zawierające mniej diod LED, niż większe wskaźniki minut. Bez diod D7...D10 jasność wyświetlaczy sekund byłaby więc znacznie większa, niż wyświetlaczy minut.

Inne obwody są identyczne, jak w wersji z małymi wyświetlaczami.

Podczas uruchomienia i testów modelu pokazanego na fotografii dały o sobie znać specyficzne cechy układu. Na pierwszy rzut oka napięcie zasilania może być dowolnie duże, a potrzebną jasność można uzyskać za pomocą rezystorów R8...R14 o odpowiedniej wartości. Okazało się jednak, że napięcie zasilające i spadek napięcia na R8...R14 nie mogą być zbyt duże. Podczas pierwszych prób przy napięciu zasilania powyżej 17V świeciły się słabo także segmenty, które powinny być wyłączone. Na katodach segmentów napięcie było na tyle wysokie, że tranzystory T5...T11 przewodziły lekko nawet wtedy, gdy na ich bazach napięcie było równe +5V. Trzeba było zmniejszyć wartość

napięcia zasilającego wskaźniki do około 15V i zmniejszyć wartości rezystorów do 22Ω, by uzyskać pożądaną jasność.

Program

Pełny listing programu umieszczony jest na stronie internetowej EdW. Program jest w sumie prosty, a plik wynikowy .bin ma tylko 619 bajtów. Realizacja jest typowa: wewnętrzny licznik Timer0 pracuje w trybie 2 i generuje przerwania podczas przepelnienia. Zastosowano kwarc o częstotliwości 8MHz, bo taki akurat był pod ręką. Wewnętrzny dzielnik podaje na Timer0 przebieg o częstotliwości $f_{osc}/12$ czyli w tym wypadku 0,666(6)MHz (okres wynosi 1,5μs). Przebieg ten trzeba podzielić za pomocą Timera0 i liczników programowych przez liczbę 666666 lub 666667 (drobna odchyłka nie ma znaczenia). Liczbę 666666 rozłożono na czynniki pierwsze i dobrano częstotliwości podziału poszczególnych liczników. Timer0 powinien pracować z jak największym współczynnikiem podziału, by w czasie między kolejnymi przepelnieniami procesor zdążył wykonać zaplanowane czynności. W trybie 2 maksymalny współczynnik podziału wynosi 255, a w proponowanym układzie wykorzystano współczynnik 222. Oznacza to, że timer generuje przerwanie co 333μs ($222 \cdot 1,5\mu s$) czyli z częstotliwością 3kHz. Można byłoby tę częstotliwość wykorzystać do sterowania wyświetlacza, ale śmiało można pracować przy mniejszej częstotliwości

i tym samym nieco mniej „męczyć” tranzystory przełączające. Dlatego przebieg z timera jest dzielony przez 7 w programowym liczniku ze zmienną T333us, co daje okres ponad 2,331ms i częstotliwość nieco mniej niż 500Hz. Każdy cykl pracy licznika T333us, powoduje obsłużenie wyświetlacza i oczywiście zostaje zliczony w kolejnych licznikach programowych ze zmiennymi T2ms (podział przez 11) i T25ms (podział przez 39). Licznik T25ms przepelnia się co jedną sekundę (teoretycznie przy kwarcu o częstotliwości idealnie równej 8MHz co 1,000001s czyli z pomijalnym błędem 0,0001%). Nie można zastosować jednego prostego licznika w miejsce liczników T2ms i T25ms, bowiem zmienne są jednobajtowe, a wymagany współczynnik podziału jest większy niż 256 i wynosi 429.

Impulsy sekundowe z licznika T25ms są zliczane przez liczniki jednostek i dziesiątek sekund i minut (*Jedsek, Dziesek, Jedmin, Dziemin*).

Przepelnienie licznika T333us, następujące co 2,331ms, powoduje wykonanie procedury obsługi wyświetlacza. Zamieszczony fragment **listingu** pokazuje niezbędne czynności. Chodzi o to by kolejno zaświecać kolejne wyświetlacze i prezentować na nich zawartość liczników sekund i minut. W danej chwili może świecić tylko jedna z czterech cyfr multipleksowanego wyświetlacza, dlatego na początku cyklu obsługi wyświetlacza wszystkie cyfry są wygaszane przez ustawienie

na liniach P3.3...P3.7 stanu wysokiego. Do sterowania potrzebny jest licznik „multipleksowy”. Do obsługi wyświetlacza czterocyfrowego wystarczy licznik liczący do 4 (0...3). Tu wykorzystano licznik *Mux* liczący do ośmiu. Takie rozwiązanie umożliwia dwukrotne zwiększenie jasności w bardzo prosty sposób – wystarczy wykorzystać stany licznika 4...7. Pomaga w tym zmienna pomocnicza *Mux1*.

Gdy biegunowość napięcia zasilającego jest „normalna”, do zmiennej *Mux1* przepisywana jest wartość licznika *Mux*. W normalnym trybie wykorzystywane są stany licznika *Mux* równe 0...3, natomiast gdy stan tego licznika wynosi 4...7, nie świeci żaden wyświetlacz.

Gdy natomiast na końcówce P3.2 pojawi się stan niski, zmienna *Mux1* zawiera wynik dzielenia bez reszty zawartości *Mux* przez dwa. Zwiększa to dwukrotnie jasność, bo wyświetlacz świeci także przy wartości licznika *Mux* 4...7. Wskazanie ma przy tym migać – realizuje to zmienna bitowa *Wygasz*. Gdy *Wygasz* ma wartość 1, program nie zaświeca żadnego wyświetlacza. Częstotliwość migania wyznacza licznik ze zmienną *Migacz* – przepełnienie tego licznika cyklicznie zmienia wartość zmiennej bitowej *Wygasz*. W proponowanym programie licznik *Migacz* pobudzany co 2,331ms liczy do 101, co daje częstotliwość migania około 2Hz.

Jeśli zmienna *Wygasz* ma wartość 0, co 2,331ms następuje też zmiana wartości zmiennych *Mux* i *Mux1* oraz zaświecenie na-

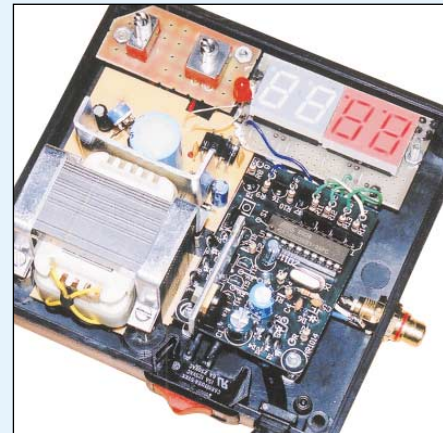
stępnego wyświetlacza. Polecenie *Select Case* sprawdza stan zmiennej *Mux1* i zaświeca jeden z wyświetlaczy, podając stan niski na jedną z końcówek P3.3...P3.7. Aby wyświetlacz pokazał odpowiednią cyfrę, do zmiennej pomocniczej *Wysw* wpisywane jest zawartość jednego z czterech liczników sekund bądź minut. Wartość zmiennej *Wysw* nie może być wyświetlona wprost, ponieważ jest to liczba dwójkowa. Aby zamienić ją na znak

Listing 1

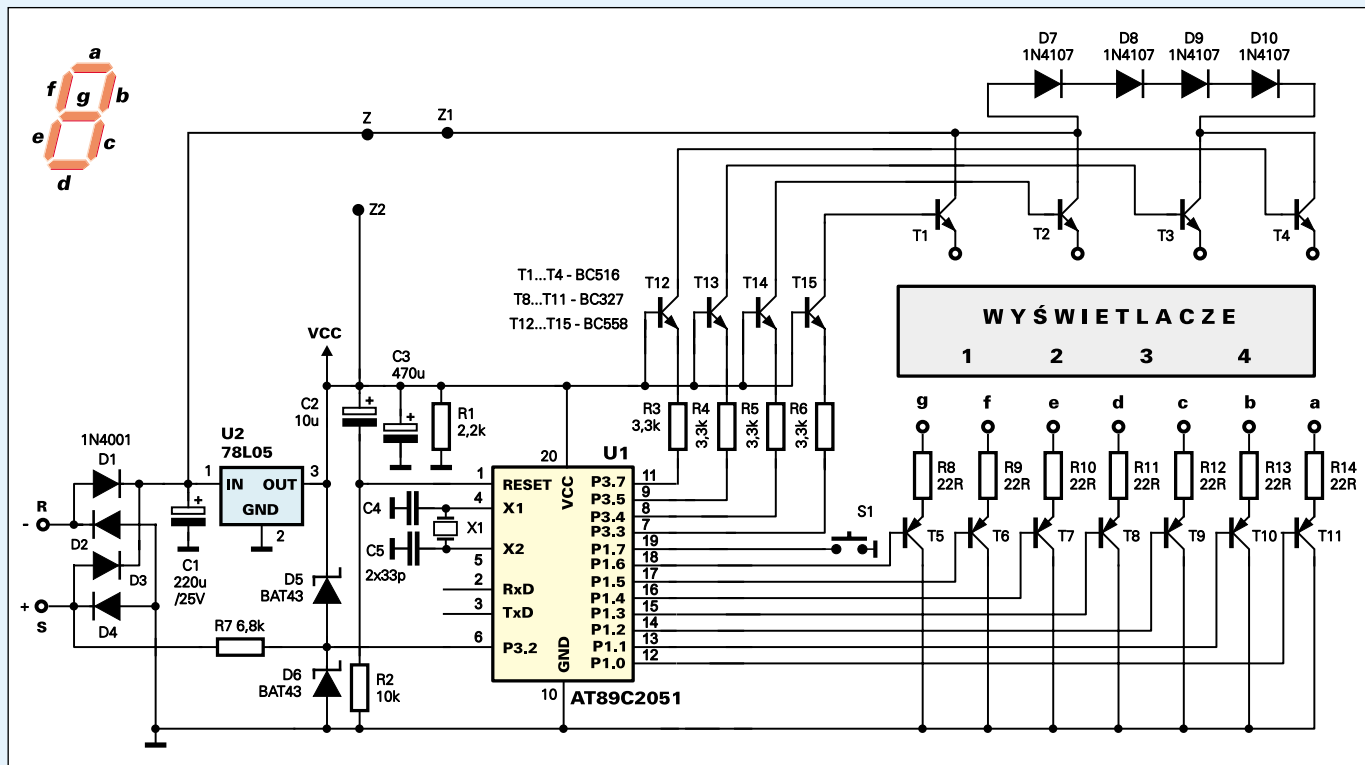
```

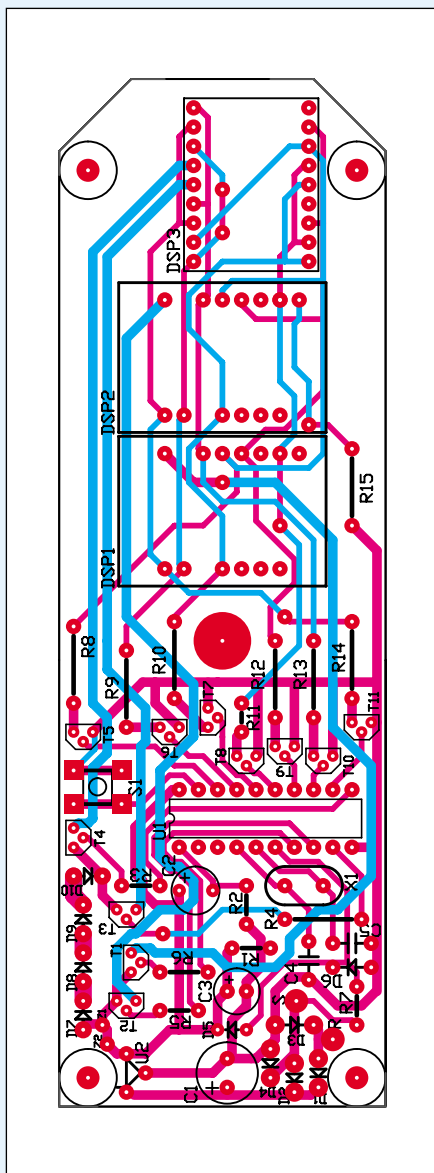
Set P3.7 : Set P3.5 : Set P3.4 : Set P3.3
Incr Mux
If Mux = 8 Then Mux = 0
Mux1 = Mux
If P3.2 = 0 Then
  Mux1 = Mux \ 2
  Incr Migacz
  If Migacz > 100 Then
    Migacz = 0
    Wygasz = Not Wygasz
  End If
End If
If Wygasz = 0 Then
  Select Case Mux1
  Case 0:
    Wysw = Jedsek
    Reset P3.7
  Case 1:
    Wysw = Dzesiek
    Reset P3.5
  Case 2:
    Wysw = Jedmin
    Reset P3.4
  Case 3:
    Wysw = Dziemin
    Reset P3.3
  End Select
End If
P1 = Lookup(wysw , Tabela)
  
```

Tabela:
Data 192 , 249 , 164 , 176 , 153 , 146 , 130 , 248 , 128 , 144



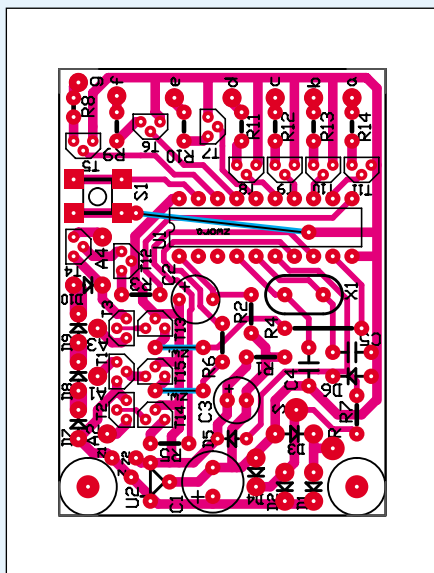
Rys. 3 Schemat ideowy wersji „JUMBO”





Rys. 4

Rys. 5



zrozumiały dla człowieka, należy zaświecić odpowiednie segmenty wyświetlacza. Konwersji z postaci dwójkowej dokonuje się dzięki poleceniu Lookup. Wartość zmiennej Wysw decyduje, która z liczb tabeli zostanie wpisana do portu P1. Jeśli przykładowo będzie wartość 0, z tabeli zostanie pobrana pierwsza liczba – 192. 192 to dwójkowo 11000000, więc na wyświetlaczu zostaną zaświecone wszystkie segmenty, z wyjątkiem segmentu g, czyli właśnie cyfra zero (pierwsza jedyńka liczby 11000000 nie ma znaczenia, bo wyprowadzenie P1.7 jest wykorzystywane do czego innego). Gdy natomiast zmienna Wysw przyjmie wartość 1, polecenie Lookup wpisze do portu P1 liczbę 249, dwójkowo 11111001, co zgodnie z oczekiwaniami zaświeci tylko segmenty b, c wyświetlacza, czyli pokaże cyfrę 1.

Montaż i uruchomienie

Wersję mniejszą według schematu z rysunku 2 można zmontować na jednostronnej płytce drukowanej, pokazanej na rysunku 4. Właśnie taka płytka i komplet elementów wchodzi w skład zestawu AVT-3012.

Płytką jest dostosowana do obudowy Z14. Pomocą przy montażu może być fotografia. Montaż jest klasyczny, najpierw warto

montować elementy najmniejsze, potem coraz większe. W modelu pod procesor dano podstawkę, pod wyświetlacze – nie. Przy budowie wersji według rysunku 2 nie powinny wystąpić żadne niespodzianki – układ od razu powinien pracować poprawnie.

Uwaga! Układ powinien być zasilany napięciem 9V z zasilacza stabilizowanego. W takim przypadku wystarczy mały stabilizator 78L05. Przy napięciu niższym niż 8,2V jasność wyświetlacza radykalnie się zmniejsza, natomiast przy napięciach powyżej 9V stabilizator U2 będzie się grzał i należałoby go zastąpić silniejszą wersją 7805 w obudowie TO-220, z ewentualnym radiatorkiem.

Nabywcy zestawu AVT-3012 otrzymają zaprogramowany mikroprocesor. Inni mogą znaleźć program napisany w BASCOM-ie na stronie internetowej EdW.

Przy testach modeli wypróbowano różne wyświetlacze i okazało się, że typowe dwucyfrowe wyświetlacze LED różnych producentów mają zdecydowanie różną sprawność, a tym samym jasność przy danym prądzie. Gdyby układ miał być zasilany z baterii, warto postarać się o wyświetlacze o podwyższonej jasności i zwiększyć wartość rezystorów R8...R14.

Ciąg dalszy na stronie 21.

Wykaz elementów

Wykaz elementów wersji małej

wg rysunków 2, 4

Rezystory

R1	2,2kΩ
R2	10kΩ
R3-R6	3,3kΩ
R7	6,8kΩ
R8-R15	10Ω

Kondensatory

C1	220μF/25V
C2	10μF/16V
C3	470μF/15V
C4,C5	33pF

Półprzewodniki

D1-D4	1N4001
D5,D6	BAT43
D7-D10	1N4107
DSP1,DSP2	wyświetlacz o wspólnej anodzie (wysokość cyfry 20mm), np. SA08-11EWA
DSP3	wyświetlacz podwójny o wspólnej anodzie (wysokość cyfry 14mm), np. DA56-11EWA
T1-T4	BC516
T5-T11	BC327
U1	AT89C2051
U2	78L05

Inne

S1	μswitch
X1	kwarc 8MHz

Wykaz elementów wersji dużej

wg rysunków 3, 5

Rezystory

R1	2,2kΩ
R2	10kΩ
R3-R6	3,3kΩ
R7	6,8kΩ
R8-R14	22Ω

Kondensatory

C1	220μF/25V
C2	10μF/16V
C3	470μF/17V
C4,C5	33pF

Półprzewodniki

D1-D4	1N4001
D5,D6	BAT43
D7-D10	1N4107
T12-T15	BC558
T1-T4	BC516
T5-T11	BC327
U1	AT89C2051
U2	78L05

Inne

S1	μswitch
X1	kwarc 8MHz
DSP1,DSP2	wyświetlacz o wspólnej anodzie (wysokość cyfry 57mm), np. SA23-11EWA firmy Kingbright
DSP3, DSP4	wyświetlacz o wspólnej anodzie (wysokość cyfry 38mm), np. SA15-11EWA firmy Kingbright

Komplet podzespołów z płytką i obudową Z14 jest dostępny w sieci handlowej AVT jako kit szkolny AVT-3012

Wersja z dużymi wyświetlaczami może być zrealizowana według rysunku 1, 3 z wykorzystaniem płytki pokazanej na **rysunku 5**. Przy zastosowaniu dwu 57-milimetrowych czerwonych wyświetlaczy SC23-11EWA oraz dwu 38-milimetrowych czerwonych wyświetlaczy SC15-11EWA - firmy King-bright - napięcie zasilania powinno wynosić około 15V. Rozkład wyprowadzeń wielu różnych wyświetlaczy LED podany był w EdW 05/2001 na stronach 41,42 i 84. Wyświetlacze wlotowane w płytkę uniwersalną i sterownik wg rysunków 3, 5 umieszczono w obudowie.

W “dużym” modelu pokazanym na fotografii wstępnej kontrolny timer z małym wyświetlaczem wbudowano w obudowę sterownika, gdzie umieszczono też zasilacz i przełączniki. Układ timera kontrolnego zrealizowano wprawdzie na płytce z rysunku 5, ale układ połączeń odpowiada rysunkowi 2. Ze względu na duże napięcie zasilania, konieczne okazało się zastosowanie stabilizatora U2 7805 w obudowie TO-220. Pracą obu timerów sterują dwa przełączniki: jeden załącz/wyłącz, drugi zmieniający biegunowość zasilania.

Piotr Górecki