

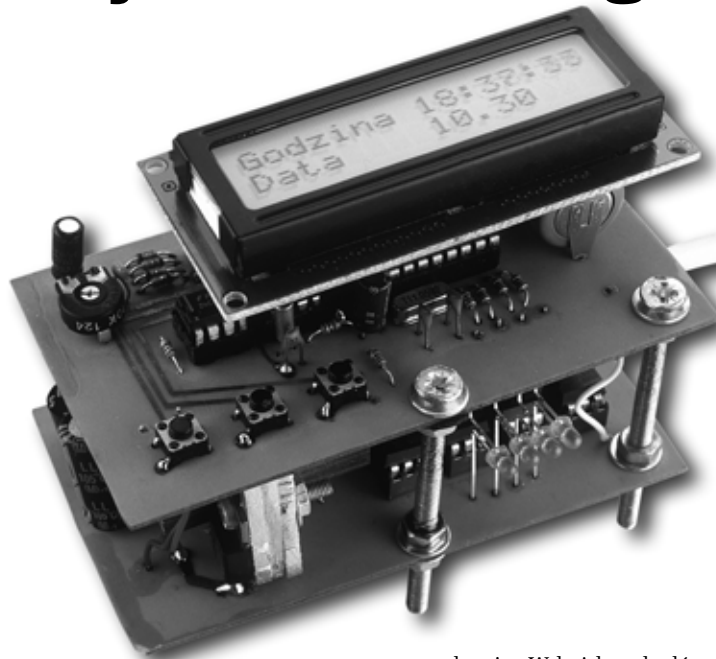
Dział „Projekty Czytelników” zawiera opisy projektów nadesłanych do redakcji EP przez Czytelników. Redakcja nie bierze odpowiedzialności za prawidłowe działanie opisywanych układów, gdyż nie testujemy ich laboratoryjnie, chociaż sprawdzamy poprawność konstrukcji.

Prosimy o nadsyłanie własnych projektów z modelami (do zwrotu). Do artykułu należy dołączyć podpisane **oświadczenie, że artykuł jest własnym opracowaniem autora i nie był dotychczas nigdzie publikowany**. Honorarium za publikację w tym dziale wynosi 250,- zł (brutto) za 1 stronę w EP. Przesyłanych tekstów nie zwracamy. Redakcja zastrzega sobie prawo do dokonywania skrótów.

Programowany sterownik zegarowy

Urządzenie przedstawione w artykule pozwala programować w trybie rocznym pracę czterech urządzeń. Dzięki wbudowanej nieulotnej pamięci nastaw, konsekwencje awarii zasilania zostały ograniczone do minimum.

Rekomendacje: urządzenie szczególnie przydatne dla śpiochów lubiących celebrować wczesnoranne wstawanie...



Projekt
111

lejnym jego atutem. Osobiście wiele razy spotkałem się z urządzeniami, których nie sposób było obsłużyć bez tytułu doktora nauk technicznych. Ponadto urządzenie jest odporne na wszelkie zaniki napięcia, ponieważ praca zegara jest podtrzymywana za pomocą akumulatora. Układ działa niezawodnie od ponad roku. Ani razu nie spóźnił się z zaparzeniem herbatki (tylko by spróbował...).

Opis układu

„Sercem” urządzenia jest mikrokontroler 89C4051 firmy Atmel (rys. 1). Ma on wbudowane 4 kB nieulotnej, reprogramowalnej pamięci Flash. W omawianym projekcie jest ona wykorzystana prawie w całości - program zajmuje 4095 bajtów tej pamięci. Kolejnymi ważnymi układami są tutaj zegar PCF8583 oraz pamięć EEPROM 24C04, komunikujące się z procesorem za pośrednictwem magistrali I²C. Do komunikacji użytkownika z urządzeniem służy wyświetlacz numeryczny (znakowy) LCD 16*2 oraz klawiatura

Z dnia na dzień staję się coraz bardziej leniwy. Takie czasy. Otóż kiedy uczęszczałem jeszcze do szkoły średniej, mieszkalem w internacie.

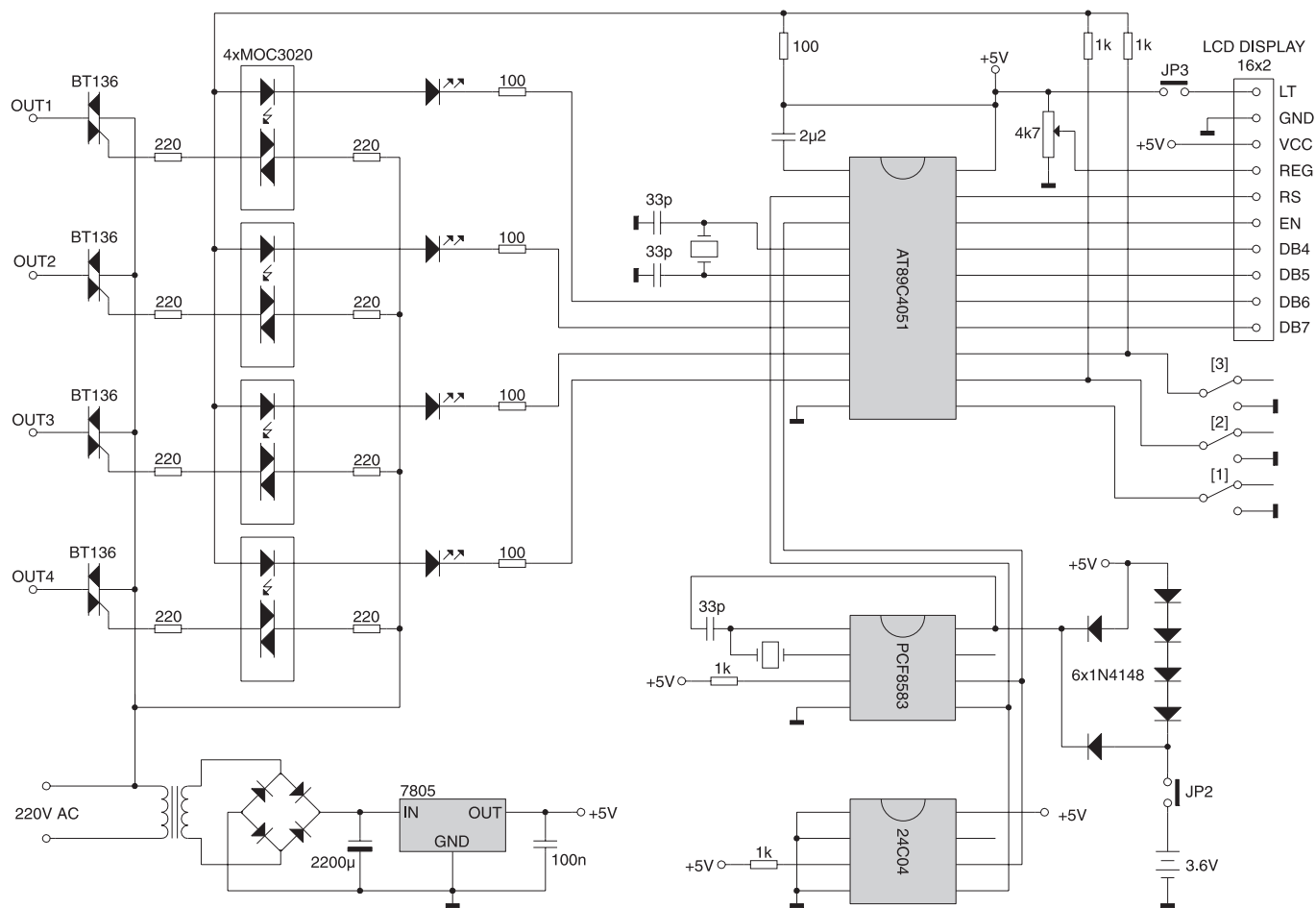
Co rano włączał się szczerze znienawidzony przeze mnie budzik, który bezlitośnie dzwoniąc, skakał po blacie mojego biurka. Kończyło się zawsze tak samo. Budzik obrywał w „czapę”, a ja powoli zwlekałem się z łóżka, ubierałem i po toalecie w lodowatej wodzie, z paskudnym humorem, kierowałem się w stronę internackiej stolówki. Tak było do czasu, kiedy to postanowiłem podnieść sobie komfort męczącej egzystencji. Lekarstwem na codzienne niewygody okazał się projekt prezentowany w artykule. Od momentu jego opracowania wszystko zmieniło się nie do poznania. Codziennie rano najpierw włącza się zaparzac do herbaty, po kilku minutach włącza się radio, a po chwili w pokoju robi się jasno. Po wypiciu szklanki gorącej herbaty, od razu nabierało się sił i ochoty do życia...

Zapewne każdy Czytelnik znajdzie jeszcze wiele innych zastosowań dla prezentowane-

go urządzenia. W każdym bądź razie zapotrzebowanie na domową automatykę istnieje i nic nie wskazuje na zmianę tej sytuacji.

Na naszym rynku dostępnych jest wiele zegarów sterujących, jednak w większości przypadków są to urządzenia przystosowane do pracy z jednym odbiornikiem. Opracowany przeze mnie sterownik nie ma tej wady, umożliwia bowiem sterowanie aż czterema niezależnymi urządzeniami (każde o mocy do 600 W), bazując przy tym na jednym elektronicznym kalendarzu. Ponadto urządzenie zapewnia niezwykle łatwą i przejrzystą obsługę, co jest ko-

Prezentowany w artykule sterownik wyposażono w programowalny zegar czasu rzeczywistego oraz kalendarz, co daje możliwość zaprogramowania pracy urządzenia w trybie rocznym. Ponadto urządzenie wyposażone jest w awaryjne źródło zasilania, co pozwala na zachowanie prawidłowego czasu i daty, nawet po odłączeniu sterownika od sieci. Informacje dotyczące załączeń/wyłączeń odbiorników przechowywane są w nieulotnej pamięci zewnętrznej typu EEPROM. W podstawowej wersji urządzenia pozwala ona na zachowanie 25 cykli załącz/wyłącz. Pamięć ustawień według producenta może być przeprogramowywana co najmniej 100000 razy, a więc urządzenie zapewnia trwałość i niezawodność oraz komfortową obsługę i eksploatację.



Rys. 1. Schemat elektryczny sterownika

zbudowana z trzech mikroprzycisków. Program główny (napisany w Bascomie) nie jest skomplikowany. Ogranicza się on do podstawowych operacji wykonywanych w nieskończonej pętli. Do operacji tych należą:

- odczyt danych z zegara (układu RTC),
- prezentacja danych na wyświetlaczu,
- sprawdzenie stanu klawiatury.

Jeżeli na ostatnim etapie nie wykryto wciśnięcia żadnego z przycisków, to program wraca do początku. W przeciwnym wypadku będą realizowane poszczególne podprogramy, po czym mikrokontroler wróci do pętli głównej.

Zarówno zasilanie awaryjne, jak i podświetlenie wyświetlacza można podłączać lub odłączać za pomocą jumperów (odpowiednio JP2 i JP3). Układ wyjściowy wykonano standardowo - elementami wykonawczymi są triaki BT136/600. Mikrokontroler steruje nimi poprzez optotriaki MOC3020 zapewniające separację galwaniczną

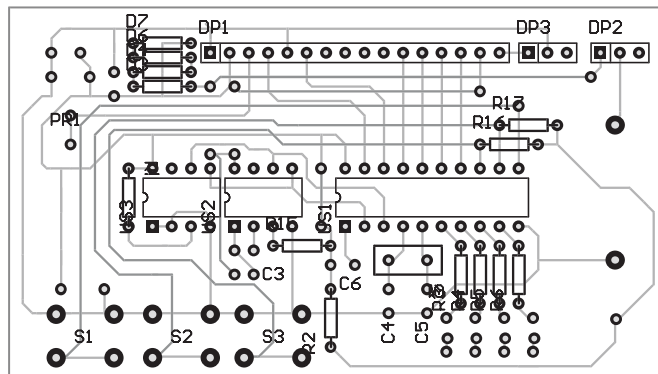
sterownika od sieci zasilającej. Zasilacz również nie jest zbyt oryginalny. Zastosowano w nim transformator TS2/16, mostek prostowniczy, stabilizator 7805 i kilka kondensatorów filtrujących.

Montaż i uruchomienie

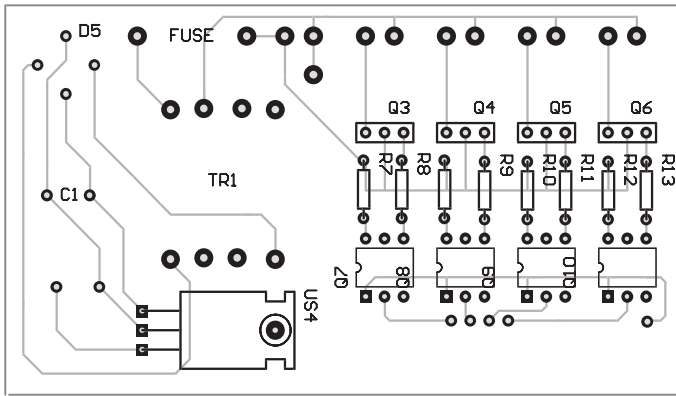
Montaż układu jest prosty i jedyny problem, zwłaszcza dla niecierpliwych wyznawców lutownicy, może stanowić sztywne umieszczenie nad sobą dwóch płytek drukowanych (schematy montażowe pokazano na rys. 2 i 3), z których sterownik się składa oraz umieszczenie ich w obudowie. Poza tym montaż nie powinien sprawić żadnych kłopotów. Jak zwykle zaczynamy od montowania podstawek, a kończymy na elementach wyższych. Pod wszystkie układy scalone zalecam stosowanie podstawek, a zwłaszcza pod procesor, gdyż jak znam życie, użytkownicy wielokrotnie będą jeszcze modyfikowali jego program. Wyświetlacz najlepiej zamontować w taki sposób, by można było później go wymienić. Najprościej uczynić to lutując do wyświetlacza lis-

twę goldpinów, a na płytce drukowanej sterownika odpowiednio gniazdo. Dwie płytki zastosowano w celu zminimalizowania wymiarów urządzenia. Z tego też powodu podjąłem decyzję o zastosowaniu laminatu dwustronnego. Na jednej z płytek znajduje się zasilacz i układ wykonawczy, na drugiej umieściłem całą „inteligencję” urządzenia, czyli mikrokontroler wraz z jego peryferiami. Pomiedzy płytkami należy w końcowej fazie montażu wykonać kilka połączeń krótkimi odcinkami dowolnego przewodu. Następnie

„poskładane” (z wlutowanymi elementami) płytki należy połączyć trzema dłuższymi śrubami, przy czym płytka zasilacza znajduje się niżej, a nad nią płytka procesora, do której to z kolei dołączamy wyświetlacz. Należy zauważyć, że na płycie układu wykonawczego nie przewidziano miejsca na bezpieczniki dla sterowanych urządzeń. Nie jest to jednak niedopatrzenie konstruktora (gdzieżby tam znowu...), ale działanie celowe. Dla wygody użytkownika urządzenia bezpieczniki te należy umieścić w gniazdach, które powinny



Rys. 2. Schemat montażowy płytki procesora



Rys. 3. Schemat montażowy płytki zasilacza

znaleźć się na ścianie obudowy. Na triaki zalecam założenie radiatora. Jeżeli chcemy użyć wyświetlacza z podświetlaniem, konieczne jest zastosowanie radiatora również przy stabilizatorze. Tak naprawdę wystarczy kawałek aluminiowej bądź miedzianej blaszki.

Procedura obsługi cykli załącz/wyłącz

Jak już wspomniano, dane dotyczące cykli załącz/wyłącz przechowywane są w nieulotnej pamięci EEPROM. Wykorzystano tutaj 250 bajtów tej pamięci. Dzieląc tę liczbę przez dziesięć, otrzymamy wynik 25 możliwych cykli załącz/wyłącz. Dlaczego tak? Otóż do prawidłowego obsługiwanie cykli (zdarzeń) potrzebne są następujące dane:

1. Godzina włączenia
2. Minuta włączenia
3. Godzina wyłączenia
4. Minuta wyłączenia
5. Dzień zdarzenia
6. Miesiąc zdarzenia
7. Numer sterowanego urządzenia

Łatwo więc zauważyć, że dane jednego cyklu zajmują 7 bajtów. Dzieląc pamięć na 25 zdarzeń, otrzymamy 10 bajtów, gdzie pozostałe trzy pozostają wolne. Może je wykorzystywać konstruktor, rozbudowując urządzenie o np. obsługę dni tygodnia itp. Za obsługę zdarzeń odpowiedzialna jest pętla główna programu, a zwłaszcza fragment przedstawiony na list. 1.

Ponieważ procedura ta wykonywana jest w pętli, wszystkie dane sprawdzane są na bieżąco. Jeżeli wszystkie warunki naszego ustawienia odpowiadają aktualnej godzinie i dacie, dochodzi do wykonania odpowiedniego zdarzenia. Niestety sprawdzenie tych wszystkich warunków, dla

wszystkich 25 ustawień, zajmuje procesorowi trochę czasu, dlatego też załączanie/wyłączanie odbywać się może z lekkim opóźnieniem (kilka sekund). Niemniej jednak, może bardziej zaawansowani ode mnie programiści poradzą sobie z tym w inny, szybszy i bardziej wydajny sposób?

Programowanie sterownika

Teraz nadszedł, zapewne wyczekiwany przez wielu konstruktorów, moment uruchomienia naszego cacka. Po podłączeniu poprawnie zmontowanego układu na wyświetlaczu pojawi się powitalny komunikat w postaci mojego adresu e-mail. Następnie układ przejdzie do wykonywania głównego programu. Na wyświetlaczu pojawi się informacja o czasie oraz dacie i o ile nie założyliśmy jeszcze odpowiedniego jumpera akumulatora, będzie to godzina 00:00:00 oraz data 01.01. Po tej niezwykle upojnej chwili wpatrywania się w wyświetlacz dobieramy się do klawiatury! Wszelkich manipulacji dokonujemy za pomocą trzech przycisków: TIME, DATE, SET.

Po naciśnięciu przycisku TIME (<) sterownik zapyta nas, czy chcemy zmodyfikować godzinę. Analogicznie po naciśnięciu przycisku DATE - zapyta, czy chcemy ustawić datę. Aby dokonać zmiany czasu albo daty, należy na zadane pytanie odpowiedzieć klawiszem SET (->). Można również odpowiedzieć negatywnie i wrócić do pętli głównej programu (opcja ESC) klawiszem: <. Przy ustawianiu zegara licznik sekund zostanie wyzerowany. Teraz możemy przejść do ustawiania godzin i minut. Służą do tego w tym momencie odpowiednio przyciski TIME i DATE. Po ustawieniu prawidłowej

informacji dokonujemy zapisu za pomocą przycisku SET. Analogicznie postępujemy podczas wprowadzania daty.

Przed dokonywaniem jakichkolwiek ustawień zalecam skasowanie pamięci. Dokonujemy tego poprzez równoczesne wciśnięcie klawiszy TIME i DATE w trybie głównym programu (podczas prezentacji godziny i daty). Następnie naciskamy SET (->). Po skasowaniu zawartości pamięci urządzenie przejdzie do trybu głównego.

Aby przejść do trybu ustawiania zdarzeń, należy pozytywnie odpowiedzieć na pytanie postawione przez sterownik po naciśnięciu klawisza SET (->). Kiedy na wyświetlaczu pojawi się napis „Następne...“, to oznacza, że w pamięci nie znajdują się dalsze/żadne ustawienia. Należy wówczas nacisnąć klawisz SET (->), po czym urządzenie przejdzie do trybu pobierania danych określonych przez nas zdarzeń. Należy teraz wpisać dane, o które poprosi sterownik. Po zakończeniu tej operacji pojawi się napis „*** OK! ***“. Teraz możemy przejść do trybu głównego. Od teraz nasze urządzenie (np. AGD) będzie obsługiwane przez nasz sterownik. Możemy również dopisywać kolejne cykle załącz/wyłącz (po naciśnięciu klawisza SET (->)). Tak wygląda tryb programowania urządzenia. Proste?

Jeżeli zdecydujemy się ponownie przejść do trybu ustawień, nie przywitana nas już napis „Następne...“. Zamiast niego pojawi się tam godzina załączenia pierwszego wprowadzonego przez nas cyklu załącz/wyłącz, a przed nią stosowny numer kolejności wprowadzenia. Teraz możemy dokonać korekty wpro-

WYKAZ ELEMENTÓW

- Rezystory**
 R1, R15...R17: 1kΩ
 R2...R6: 100Ω
 R7...R14: 220Ω
 PR1: potencjometr napiłtkowy miniaturowy 4,7kΩ
- Kondensatory**
 C1: 2200µF/25V
 C2: 100nF
 C3...C5: 33pF
 C6: 2,2µF/16V
- Półprzewodniki**
 D1...D4, D6, D7: 1N4148
 D5: mostek prostowniczy 1A/50V
 U1: AT89C4051 zaprogramowany
 U2: PCF8583
 U3: AT24C04
 U4: 7805
 Q3...Q6: BT136
 Q7...Q10: optotriak MOC3020
- Różne**
 Q1: rezonator kwarcowy 11,059 MHz
 Q2: rezonator kwarcowy zegarkowy 32768 Hz
 TR1: transformator TS2/16
 FUSE 1: bezpiecznik 100 mA
 FUSE 2...5: bezpiecznik 3 A
 DP1: wyświetlacz LCD 16*2
 JP2, JP3: goldpin 3*1
 S1-S3: przycisk typu microswitch
 Bat 1: akumulator 3,6V

wadzonych przez nas uprzednio danych (klawisz SET (->)). Wybierając opcję NEXT (klawisz DATE), możemy natomiast przejść do trybu wprowadzania kolejnych zapisów.

Teraz jeszcze jedna bardzo ważna sprawa odnośnie wpisywania cykli załącz/wyłącz. Mianowicie bardzo często za pomocą nieskomplikowanego triku programistycznego moż-

List. 1. Obsługa cykli załącz/wyłącz

```

Licznik = Licznik + 10
X = Licznik + 1 : Call Read_eeprom(x, Value)
If Minuty = Value Then
X = Licznik + 2 : Call Read_eeprom(x, Value)
If Godziny = Value Then
X = Licznik + 3 : Call Read_eeprom(x, Value)
If Dni = Value Or Value = 32 Then
X = Licznik + 4 : Call Read_eeprom(x, Value)
If Miesiace = Value Or Value = 13 Then
X = Licznik + 9 : Call Read_eeprom(x, Value)
Call Wlacz_urzadzenie
End If
End If
End If
End If

X = Licznik + 5 : Call Read_eeprom(x, Value)
If Minuty = Value Then
X = Licznik + 6 : Call Read_eeprom(x, Value)
If Godziny = Value Then
X = Licznik + 9 : Call Read_eeprom(x, Value)
Call Wylacz_urzadzenie
End If
End If

If Licznik = 250 Then Licznik = 0
    
```

na w znaczny sposób rozszerzyć funkcjonalność danego urządzenia mikroprocesorowego. Tak też stało się i tym razem. Tryb roczny pozwala na bardzo precyzyjne sterowanie urządzeniami. Jednak przyznacie, że w przypadku, gdy chcemy, aby dane zdarzenie zachodziło codziennie o takiej samej godzinie, bezcelowym i zapewne bardzo męczącym byłoby wprowadzanie wszystkich dni roku. Zresztą dni mamy 365, a ustawień tylko 25. Ponadto może również zaistnieć konieczność włączania danego urządzenia takiego samego dnia każdego miesiąca albo odwrotnie, kiedy chcemy, aby urządzenie włączało się codziennie podczas wybranego miesiąca. Nasz sterownik umożliwia takie triki! Być mo-

że podczas wprowadzania cykli załącz/wyłącz natknąłeś się, Czytelniku, na dwa plusy „++”, które pojawiają się po przekroczeniu dnia 31 oraz po przekroczeniu 12 miesiąca. Te plusy oznaczają, że ten argument będzie ignorowany. Łatwo więc zauważyć, że w przypadku ignorowania dni oraz ignorowania miesięcy zdarzenie odbywać się będzie codziennie, w przypadku ignorowania dni - cykle wykonywane będą codziennie, oraz w przypadku ignorowania miesięcy - co miesiąc danego dnia.

Aktualizacja oprogramowania

Jeżeli zainteresowanie układem będzie duże, a programiści aktywni, założę stronę WWW sterownika,

gdzie zamieszczane będą kolejne wersje oprogramowania (wraz z danymi o autorach oczywiście). Jasnym jest chyba fakt, że cała „inteligencja” zależy od programu procesora. Sprzęt nie gra tu wiążącej roli, zwłaszcza jeżeli chodzi o przeznaczenie układu. Będę więc czekał na wszelkie propozycje i rozwiązania programowe nowych zastosowań urządzenia. Niestety z bólem serca muszę stwierdzić, że te 4 kB pamięci to trochę mało (dla mnie i Bascoma). Może uda Wam się znacznie ograniczyć długość kodu poprzez wykorzystanie innego kompilatora (np. języka C, ewentualnie ASM)? Przyznaję, że program, który udostępniłem, to nie jest artystyczne dzieło, ale wystar-

cza do prawidłowego funkcjonowania urządzenia. Jednak całą zabawę związaną z sortowaniem pozostawiam Wam, drodzy Czytelnicy. Miłej zabawy!

Mariusz Ciszewski
elemid@wp.pl

Od redakcji: przygotowane przez autora projektu płytki drukowane wymagają modyfikacji polegającej na pogrubieniu ścieżek doprowadzających prąd do obciążenia, warto także pogrubić do co najmniej 20...30 milów pozostałe ścieżki - ułatwi to wykonanie płytek w warunkach domowych. Tak zmodyfikowane wzory płytek publikujemy na wkładce i na CD-EP7/2003B.