



Supergadżet Świetlny



Opisany poniżej układ prezentuje bardzo ciekawy efekt świetlny. Można znaleźć wiele zastosowań. Znakomicie nadaje się do oświetlenia akwarium. Kolorowe, migoczące światło diod odbija się od błyszczących, poruszających się rybek, dając piękny, niezwykły efekt. Równie dobrze może posłużyć jako oświetlenie wymarzonych prezentów oczekujących na otwarcie pod wspaniałą, zieloną, pachnącą choinką w czasie świąt. Oprócz tego układ świetnie nadaje się do

zastosowania na dyskotecze i może z powodzeniem wspomagać profesjonalne urządzenia.

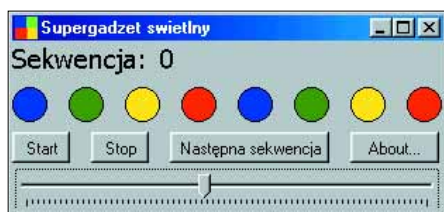
Układ zbudowany jest ze sterownika i dwóch modułów kolorowych diod (w podstawowej wersji). Każdy moduł ma osiem diod, które mogą migać w dziewięciu różnych sekwencjach (1-9). Dodatkowo sekwencja 0 składa się z wybranych przez nas pozostałych sekwencji 1-9, które są wyświetlane po kolei. Następnym atutem podnoszącym funkcjonalność gadżetu jest możliwość zmiany szybkości migotania diod. Ogromną zaletą układu jest pamiętanie ustawień po wyłączeniu zasilania. Dzięki temu nie trzeba konfigurować układu przy każdym uruchomieniu. Moduły diod są połączone ze sterownikiem i między sobą

równolegle. Pozwala to na połączenie ich większej liczby.

Sposób migania diod w poszczególnych sekwencjach można prześledzić w specjalnie napisanym do tego celu programie SGS.exe (Supergadżet świetlny), który jest umieszczony na stronie internetowej EdW. Wygląd okna programu pokazany jest na **rysunku 1**. Należy jednak zaznaczyć, że jest to tylko namiastka rzeczywistego efektu, mająca dać wstępny obraz działania gadżetu. Układ mimo stosunkowo prostej budowy posiada kilka ciekawych rozwiązań, o których warto przeczytać w dalszej części artykułu.

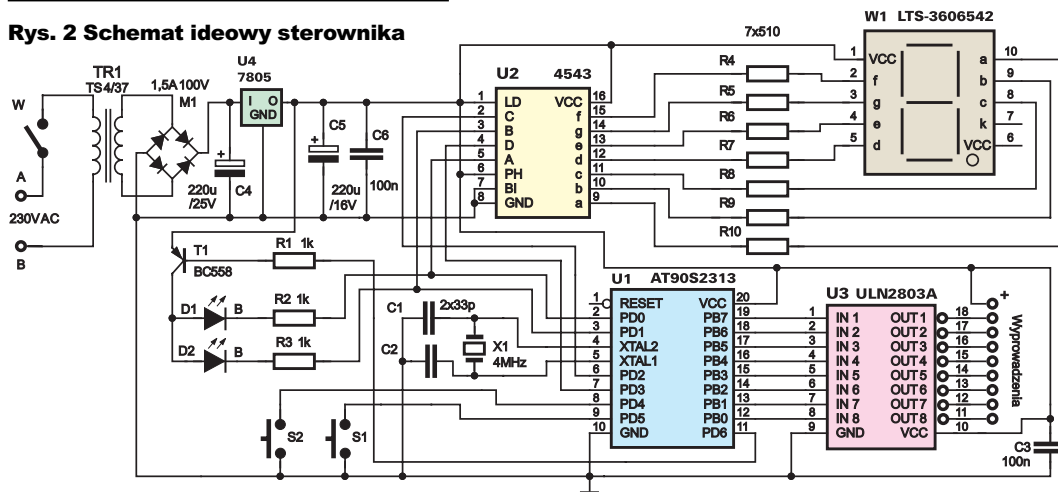
Opis układu

Schemat ideowy sterownika pokazany jest na **rysunku 2**, schematy modułów diod przedstawiono na **rysunkach 3, 4**. Jak widać, układ

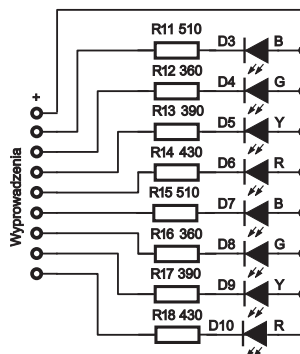


Rys. 1 Wygląd okna programu SGS.exe

Rys. 2 Schemat ideowy sterownika



Rys. 3 Schemat ideowy modułu diod 1



nie jest skomplikowany. Pracą urządzenia steruje mikrokontroler 90S2313 (U1). Układ ULN2803A (U3) jest scalonym driverem mocy. Pełni funkcję „pomostu” między mikroprocesorem a diodami. Zawiera 8 par tranzystorów NPN typu darlington. Katalog podaje, że prąd jego jednego wyjścia może wynosić nawet 500mA. Można więc podłączyć do niego naprawdę sporą liczbę diod. U2 (4543) steruje wyświetlaczem W1 – zamienia kod binarny podany z procesora na siedmiosegmentowy. Wyświetlacz pokazuje numer sekwencji lub numer szybkości migania diod w zależności od trybu.

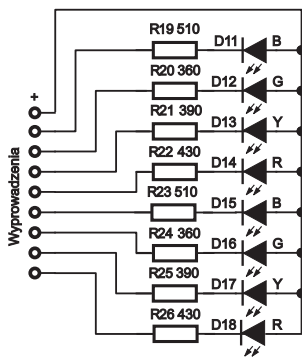
W zasilaczu zastosowałem dobrze znaną kostkę 7805 (U4). Razem z transformatorem jest częścią gadżetu, więc można go zasilać bezpośrednio z sieci energetycznej.

Program sterujący został napisany w Basicie i zajmuje prawie całą pamięć procesora. Jego działanie polega na ciągłym wywoływaniu podprogramów wyświetlających po-

szczególne sekwencje. Pętla główna pokazana jest na **listingu 1**. Przykład podprogramu odpowiedzialnego za jedną z sekwencji jest przedstawiony na **listingu 2**.

Układ sterownika pracuje w trzech trybach (0, 1, 2), których opis znajduje się w **tabeli 1**. Tryby można zmieniać za pomocą S1 (Tryb/Kasuj). Diody D1, D2 informują, w jakim trybie znajduje się układ. Uważny Czytelnik zapyta po co im jest potrzebny tranzystor T1. Otóż w trybach 1 i 2 mikrokontroler steruje pracą tych diod i wyświetlacza w sposób multipleksowy. Dzieje się to za pomocą linii PD0, PD1, PD2, PD3. No tak, ale jak to możliwe, skoro zasilanie W1 jest podłączone na stałe? Skorzystałem tutaj z bardzo pożytecznej właściwości układu U2 (4543). Mianowicie, jeśli podamy na jego wejścia liczbę większą od 9, to na wszystkich wyjściach pojawia się stan wysoki, wyświetlacz nie świeci. W tym momencie można sterować diodami D1, D2. Na przykład, jeżeli na liniach PD0, PD1, PD2, PD3 pojawi się liczba 13 (B1101), to zaświeci się dioda D2, a wyświetlacz zgaśnie. Natomiast gdy będzie to liczba mniejsza lub równa 9, wyświetlacz ją pokaże, ale diody D1, D2 nie będą świecić, bo wtedy wyłączymy tranzystor T1. Odpowiedzialny za to fragment programu jest pokazany na **listingu 3**.

Rys. 4 Schemat ideowy modułu diod 2



Tryb	Diody D1, D2	Opis
0	Nie świecą	Układ steruje diodami w jednej z 10 sekwencji
1	Świeci D1	Zmiana szybkości migania diod
2	Świeci D2	Programowanie sekwencji 0

Tabela 1

Tabela 2

Tryb	Funkcja S1(Tryb/Kasuj)	Funkcja S2(Zwiększ/Ustaw)
0	Przejdź do trybu 1	Zmiana sekwencji migania diod
1	Przejdź do trybu 2	Zmiana szybkości migania diod
	Przejdź do trybu 0	Dalej
2	Usunięcie sekwencji pokazanej na wyświetlaczu z sekwencji 0	Wstawienie sekwencji pokazanej na wyświetlaczu do sekwencji 0

Numer szybkości	Orientacyjny czas opóźnienia [ms]
0	150
1	300
2	450
3	600
4	750
5	900
6	1050
7	1200
8	1350
9	1500

Tabela 3

tym, że diody informacyjne D1, D2 nie świecą. Pozostałe diody migają w wybranej sekwencji, a jej numer (0-9) jest pokazany na wyświetlaczu. Sekwencje zmieniamy za pomocą S2 (Zwiększ/ Ustaw).

Tryb 1 służy do zmiany szybkości migania diod. Diody mogą migać z dziesięcioma ponumerowanymi (0-9) szybkościami, które zmieniamy również za pomocą S2 (Zwiększ/ Ustaw). Na wyświetlaczu w tym trybie cały czas pokazany jest numer szybkości i świeci D1. Orientacyjne czasy opóźnień zamieszczone są w **tabeli 3**.

W trybie 2 możemy zaprogramować sekwencję 0, w której wyświetlane są wybrane przez nas pozostałe sekwencje po kolei. Po wejściu do tego trybu zaświeci się dioda D2, a na wyświetlaczu pojawi się „0”. Jeżeli teraz naciśniemy S1 (Tryb/Kasuj), to wyjdziemy

z trybu 2. Układ obsługujemy za pomocą przycisków S1 (Tryb/Kasuj), S2 (Zwiększ/ Ustaw). Ich funkcje zestawione są w **tabeli 2**.

Tryb 0 jest trybem głównym. Można go poznać po

Listing 1

```

Startt:          'pętla główna
Select Case Sekw
Case 0 : Call Sekw0 'Zmienna Sekw przechowuje numer
Case 1 : Call Sekw1 'aktualnie wybranej sekwencji
Case 2 : Call Sekw2
Case 3 : Call Sekw3
Case 4 : Call Sekw4
Case 5 : Call Sekw5
Case 6 : Call Sekw6
Case 7 : Call Sekw7
Case 8 : Call Sekw8
Case 9 : Call Sekw9
End Select
Goto Startt      'koniec pętli głównej
    
```

Listing 2

```

' Przykład podprogramu wyświetlającego sekwencję 2
Sub Sekw2
Portb = &H88 : Call Opoz
Portb = &H44 : Call Opoz
Portb = &H22 : Call Opoz
Portb = &H11 : Call Opoz
Portb = &H22 : Call Opoz
Portb = &H44 : Call Opoz
If Sekw = 0 Then 'warunek potrzebny gdy Sekw2
jest częścią sekwencji 0
Portb = &H88 : Call Opoz
End If
End Sub
    
```

Listing 3

```

'Multipleksowa praca D1, D2 i wyświetlacza
'część procedury Opoz
'Tryb 1
Toggle Flag      'zanegowanie flagi
If Flag = 0 Then
Portd = &B0111110 'zgaś wyświetlacz, włącz tranzystor, zapal D1
Else
Portd.6 = 1      'wyłącz tranzystor
Portd = Szyb Or &HF0 'pokaż numer szybkości na wyświetlaczu
End If           'koniec if flag
...
...
...
'Tryb 2 (podprogram Multi)
Toggle Flag      'zanegowanie flagi
If Flag = 0 Then
Portd = &B0111101 'zgaś wyświetlacz, włącz tranzystor, zapal D2
Else
Portd.6 = 1      'wyłącz tranzystor
Portd = Try Or &HF0 'pokaż numer sekwencji na wyświetlaczu
End If           'koniec if flag
    
```

z tego trybu i wrócimy do trybu głównego – 0. Natomiast jeśli naciśniemy S2 (Zwiększ/Ustaw), na wyświetlaczu pojawi się „2” i diody w modułach przestaną migać. Oznacza to, że układ pyta nas: czy chcemy, żeby sekwencja 2 była częścią sekwencji 0. Jeżeli tak, to naciskamy S2 (Zwiększ/Ustaw). W przeciwnym razie S1 (Tryb/Kasuj). Na wyświetlaczu pojawi się „3”. Tym razem układ pyta, czy chcemy, aby sekwencja 3 pojawiła się w sekwencji 0. I znowu naciskamy jeden z klawiszy: S1 (Tryb/ Kasuj) – Nie, S2 (Zwiększ/Ustaw) – Tak. Układ zapyta o następną sekwencję – 4. I tak do 9. Po udzieleniu odpowiedzi na temat sekwencji 9 następuje automatycznie przejście do trybu głównego – 0. Ktoś mógłby zapytać, dlaczego pomijana jest sekwencja 1 – układ zaczyna pytać od sekwencji 2. Chodzi o to, żeby nie było takiej sytuacji, że w sekwencji 0 nic nie zostanie. Jeżeli wykasujemy wszystkie sekwencje (2-9), to zostanie sekwencja 1, w której wszystkie diody świecą.

Obsługa przycisków S1, S2, a co za tym idzie komunikacji użytkownika z układem jest zawarta w opóźnieniu (*Opoz*) między kolejnymi „mignięciami” diod. Cała ta procedura jest pętlą *For*. Jej fragment jest pokazany na **listingu 4**. Liczba cykli tej pętli zależy od wybranej szybkości migania diod – zawartości zmiennej *Opoz1*. Jeden jej cykl trwa, licząc z przybliżeniem ok. 1ms. Więc aby uzyskać przerwę między kolejnymi „mignięciami diod” rzędu, powiedzmy 150ms, należy tę pętlę powtórzyć 150 razy. Na tym listingu jest również pokazany algorytm nietypowego opóźnienia po wciśnięciu przycisków. Nie wszędzie w programie dało się zastosować prostą komendę *waitms 250*. W trybie 1 i 2 zakłócałoby to multipleksowe sterowanie diodami informacyjnymi D1, D2 i wyświetlaczem, które wchodzi w skład procedury *Opoz*. Po naciśnięciu przycisku jedna z diod informacyjnych (lub wyświetlacz) gasłaby na te 250ms. Program z listingu 3 byłby wstrzymany. Dlatego, jak widać, zastosowano tutaj pomocniczą zmienną *Zm*. Jest ona zwiększana przy wciśniętym przycisku w każdym cyklu pętli *For*. Dopiero gdy osiągnie pewną wartość (90), następuje właściwa obsługa przycisku.

Ponieważ na zmiennych zapisanych w pamięci EEPROM nie można wykonywać operacji arytmetycznych, a taką jest zmienna *Szyb*, wykorzystano do tego zmienną *Zm*. (Zmienna *Szyb* przechowuje numer szybkości migania diod).

Jak wspomniano na początku, układ pamięta ustawienia po wyłączeniu zasilania. Należą do nich: aktualnie wyświetlana sekwencja, szybkość migania diod oraz zawartość sekwencji 0. Ustawienia są przechowywane oczywiście w pamięci EEPROM mikrokontrolera. Dla niewtajemniczonych: aby można było jakąś zmienną zapisać w pamięci EEPROM, wystarczy tylko dodać słowo *ERAM* przy jej deklarowaniu. Na przykład: *Dim*

Listing 4

```
Sub Opoz      'podprogram opóźnienia między "mignięciami diod"
For Zm2 = 0 To Opoz1 'liczba powtórzeń pętli zależy od wybranej szybkości
migania
...          'zmienna Opoz1 określa czas opóźnienia
...
...
If Tryb = 1 Then 'zmiana szybkości migania diod
If Pind.4 = 0 Then 'jeżeli wciśnięty S2
Incr Zm          'zwiększ pomocniczą zmienną
If Zm >= 90 Then 'jeżeli czas wciśnięcia przycisku jest odpowiednio długi to
Flag2 = 0       'wyzeruj flag2
Zm = Szyb       'zwiększ numer szybkości migania diod korzystając
Incr Zm         'ze zmiennej pomocniczej Zm
Szyb = Zm
Zm = 0
If Szyb = 10 Then
Szyb = 0 : Opoz1 = 0
End If          'koniec if szyb = 10
Opoz = Opoz1 + 150 'zwiększ opóźnienie między "mignięciami diod" o 150ms
Opoz1 = Opoz
End If         'koniec if zm>90
End If
...
...
'nie zmieniaj szybkości o więcej niż o 1 jeżeli przycisk jest cały czas wciśnięty
If Flag2 = 0 And Pind.4 = 0 Then Zm = 0
...
...
'jeżeli przyciski nie są wciśnięte to ustaw flagę pomocniczą flag2
If Pind.4 = 1 And Pind.5 = 1 Then Flag2 = 1
Waitms 1
Next
...
...
End Sub      'koniec podprogramu Opoz
```

Zmienna as Eram Byte.

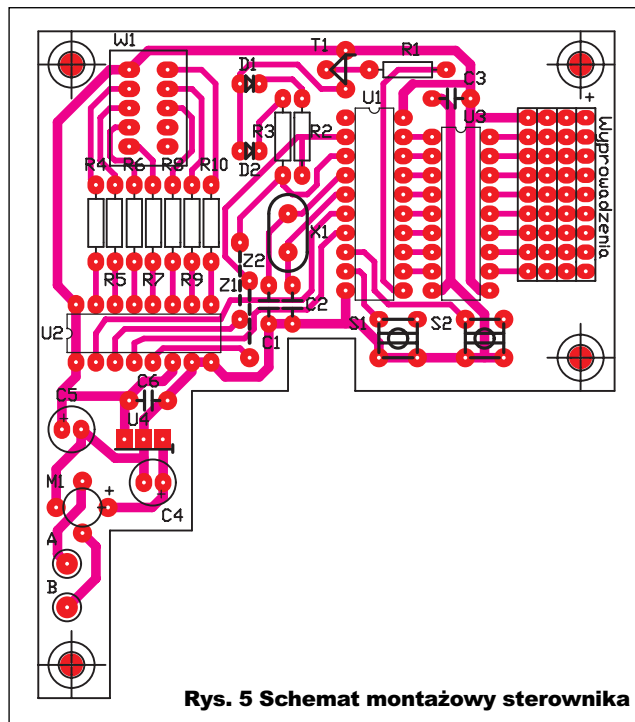
Jeżeli ktoś jest zainteresowany dalszymi szczegółami dotyczącymi programu, to odsyłam do pełnego kodu źródłowego umieszczonego w Przyjaznym Portalu Elektroniki dla Wszystkich. Tyle na temat działania układu. Przechodzimy do kwestii jego wykonania.

Montaż i uruchomienie

Schematy montażowe sterownika, modułu diod 1, modułu diod 2 są przedstawione na **rysunkach 5, 6 i 7**. Płytkę sterownika zaprojektowano tak, aby zmieściła się razem z transformatorem do obudowy Z5A.

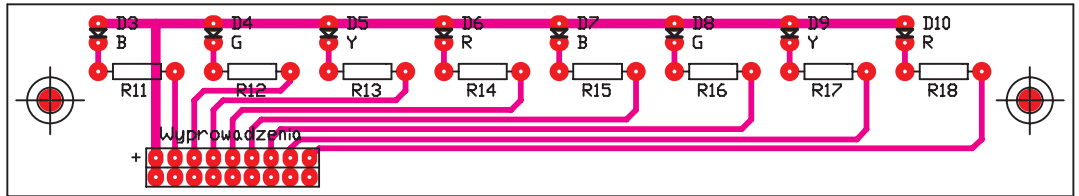
Montaż należy rozpocząć od wlotowania dwóch zwór Z1, Z2. Dalej elementy montujemy klasycznie od najmniejszych do największych. Dobrym zwyczajem jest umieszczanie układów scalonych w podstawkach, zwłaszcza mikrokontrolera, gdyż jest to stosunkowo kosztowny element. Jeżeli ktoś będzie chciał umieścić układ w obudowie, to trze-

ba przedłużyć nóżki elementów S1, S2 i wyświetlacza. W module diod najlepiej zastosować diody o podwyższonej jasności świecenia, bo właśnie do nich zostały dobrane rezystory ograniczające. Oprócz tego dają najlepszy efekt. W większości diod dłuższa nóżka ozna-



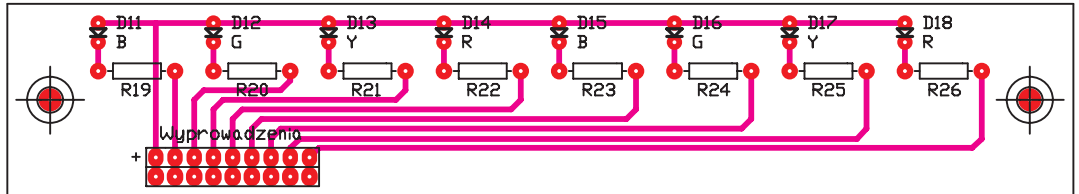
Rys. 5 Schemat montażowy sterownika

cza anodę, ale zdarzają się egzemplarze, w których jest na odwrót. Dlatego warto sprawdzić diody za pomocą miernika przed ich wlutowaniem. Jak już wcześniej wspomniano, moduły diod i sterownik łączymy równolegle. Każdy moduł ma po dwa rzędy wyprowadzeń. Umożliwia to „szeregowe” ich połączenie (bezpośrednio ze sobą). Najlepiej nadaje się do tego taśma komputerowa. Dzięki niej unikamy płątaniny przewodów i całość wygląda porządnie i estetycznie. Trzeba tylko zwrócić uwagę, aby wlutować taśmę w dobrą stronę – plus do plusa. Na obudowę można przykleić elegancką naklejkę, która jest pokazana na rysunku 8. Przewidziano w niej miejsce na wyłącznik sieciowy.



Rys. 6 Schemat montażowy modułu diod 1

Rys. 7 Schemat montażowy modułu diod 2



Rys. 8 (skala 50%) Naklejka

Po sprawdzeniu poprawności montażu można włączyć zasilanie. Układ wejdzie w tryb 0. Diodы informacyjne D1, D2 nie świecą. Na wyświetlaczu pokaże się numer sekwencji. Przy pierwszym włączeniu diody

w modułach będą migać w sekwencji 0 z szybkością numer 4. Sekwencja 0 składa się ze wszystkich pozostałych.

Możliwości zmian

Płytkę sterownika ma 4 rzędy wyprowadzeń, dlatego zgodnie z tym co napisano na początku, układ można łatwo rozbudować przez dodanie modułów diod. Trzeba tylko wziąć pod uwagę wydajność prądową stabilizatora (1A) oraz transformatora – w tym przypadku 300mA. Może również zaistnieć potrzeba przykręcenia niewielkiego radiatora do stabilizatora, jeżeliby się nadmiernie grzał. W naklejce przewidziano otwór na wyłącznik sieciowy, ale również dobrze można go zamontować na kablu. Wtedy należałoby nieco zmienić projekt naklejki. W podstawowej wersji układu diody w modułach ułożone są w ten sam sposób. Jednak dlaczego by nie spróbować innego rozmieszczenia? Można na przykład zrobić kilka modułów diod i w każdym z nich zastosować diody o jednym kolorze.

Efekt może okazać się równie ciekawy. Możliwości jest wiele. Wszystko zależy od pomysłowości wykonawcy i od tego, gdzie ma być zastosowany układ.

Paweł Karcz
dekon@interia.pl

Wykaz elementów

Rezystory

R1-R3	1kΩ
R4-R10,R11,R15	510Ω
R12,R16,R20,R24	360Ω
R13,R17,R21,R25	390Ω
R14,R18,R22,R26	430Ω
R19,R23	510Ω

Kondensatory

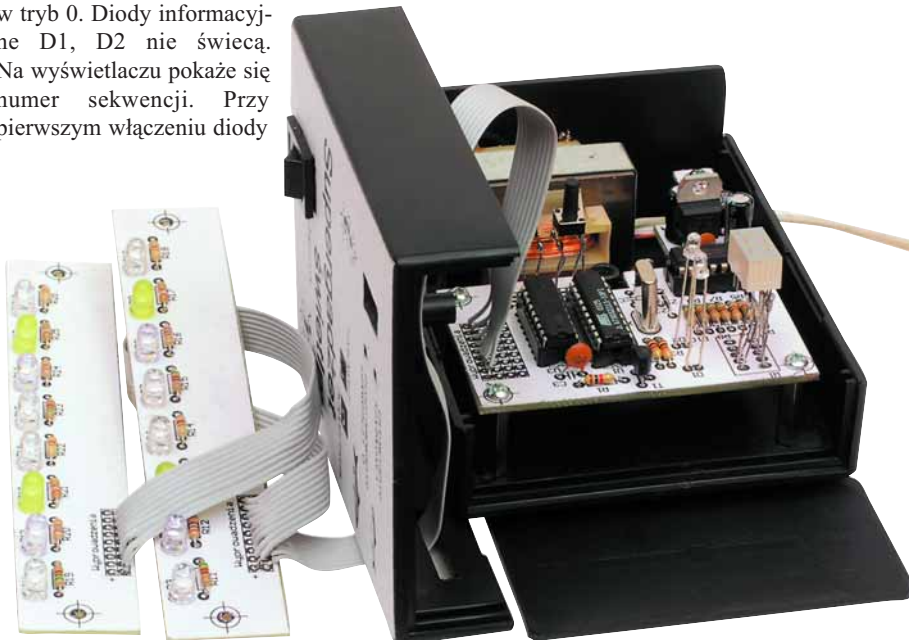
C1,C2	33pF
C3,C6	100nF
C4	220μF/25V
C5	220μF/16V

Półprzewodniki

D1,D2,D3,D7,D11,D15	LED 5mm B
D4,D8,D12,D16	LED 5mm G
D5,D9,D13,D17	LED 5mm Y
D6,D10,D14,D18	LED 5mm R
M1	1,5A 100V
T1	BC558
U1	AT90S2313
U2	4543
U3	ULN2803A
U4	7805

Pozostałe

TR1	TS 4/37
W1	LTS-3606 542
X1	rezonator kwarcowy 4MHz



Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-2758.