

Sterownik węża świetlnego na EPROMie

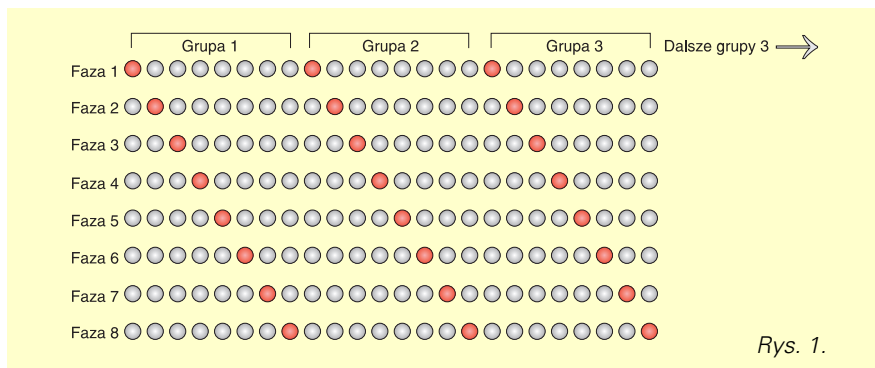
Podobnie jak wiele układów z serii 2000, proponowane urządzenie służy zabawie i rozrywce, ale może znaleźć także bardziej poważne zastosowania, np. w reklamie czy dekoracji witryn sklepowych z okazji świąt. Oczywiście, jednym z najbardziej typowych zastosowań węża jest oświetlenie choinki, temat już wielokrotnie poruszany w EdW.

Z pewnością większość Czytelników wie, jak wygląda znany od dawna i popularny układ węża świetlnego. Jest to po prostu szereg punktów świetlnych, lampek lub diod LED, składający się z wielu równoległe ze sobą połączonych grup. Odpowiadające sobie punkty z każdej grupy możemy zapalać w odpowiedniej kolejności, uzyskując efekt "przesuwania się" węża świetlnego. Po co resztą pisać o czymś, co najlepiej pokazać na rysunku? **Rysunek 1** najlepiej ilustruje zasadę działania naszego świecącego gada. Jest to najprostszy z wielu możliwych układów świetlnych, które możemy zrealizować i zrealizujemy.

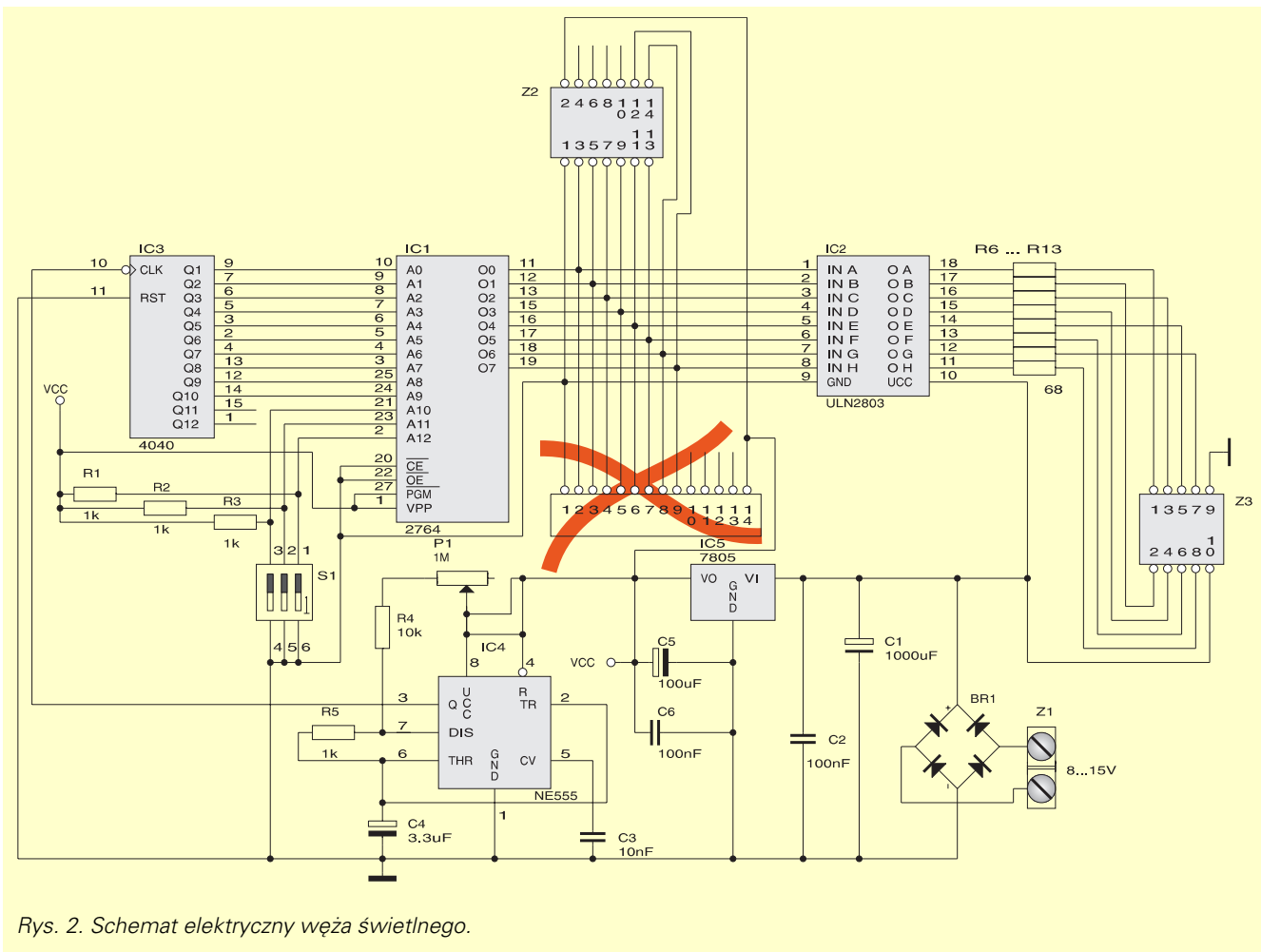
Skonstruowanie układu elektronicznego mającego sterować punktami świetlnymi węża z pozoru wydaje się proste. Wystarczy przecież do wyjść dowolnego licznika Johnsona (np. 4017)

dołączyć odpowiednie drivery i po kłopotach! No tak, ale może zechcemy aby wąż poruszał się w dwie strony? Też niewielki problem: licznik rewersyjny i dekodery 1 z 10 (np. 40192 i 4028). Ludzka wyobraźnia i zachłanność nie mają granic i tym razem może zamarzyć się nam wyświechtanie poruszających się dwóch punktów naraz, a może nawet trzech lub więcej. A może cały szereg światełek powinien w określonych momentach migotać? To wszystko dałoby się oczywiście zrealizować za pomocą typowych układów logicznych, ale stopień komplikacji układu przekroczyłby z pewnością granice zdrowego rozsądku. Postąpimy więc inaczej: wszystkie kombinacje logiczne potrzebne do uzyskiwania dowolnych efektów świetlnych zapiszemy w pamięci reprogramowalnej EPROM, a cała pozostała część układu zostanie maksymalnie uproszczona. Pomysł okazał się doskonały: jak za chwilę zobaczycie proponowany układ składa się z zaledwie pięciu układów scalonych, wliczając w to stabilizator napięcia i driver zasilający diody świecące!

Wielkim atutem proponowanego układu jest możliwość jego współpracy z innymi układami z serii 2000, a konkretnie z modułami wykonawczymi AVT-2099, AVT-2098 i z modułem wykonawczym na triakach AVT-2097, będącym obecnie w fazie testowania przed publikacją jego opisu.



Rys. 1.



Rys. 2. Schemat elektryczny węża świetlnego.

Opis działania

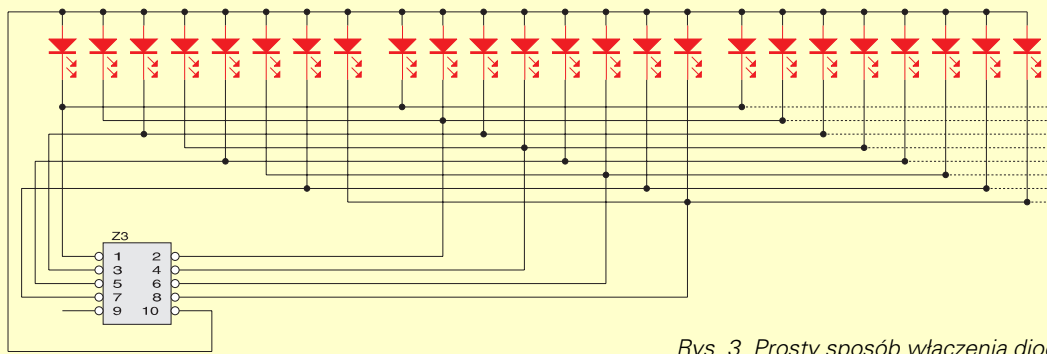
Schemat elektryczny sterownika węża świetlnego przedstawiony został na **rysunku 2**. Zanim jednak przejdziemy do analizy zasady działania układu, musimy się jeszcze rozprawić z pewnym problemem, który od dawna spędza autorowi sen z powiek. Otóż w dawnych czasach, w pionierskim okresie wydawania EdW doszło do drobnej pomyłki. Nie przerażajcie się, nic wielkiego, wszystkie układy działają! Opracowując schemat pierwszego układu z serii robotyki autor przez rozróżnienie umieścił na schemacie zamiast symbolu złącza 2x7 goldpin, symbol złącza jednorzędowego 14 goldpin. Płytkę została oczywiście zaprojektowana poprawnie i dopiero po pewnym czasie zauważono pomyłkę. Ponieważ jednak w międzyczasie opublikowane zostały inne układy z tej serii, których schematy zawierały tę samą nieścisłość, postanowiliśmy niczego nie zmieniać, aby nie narobić jeszcze większego bałaganu. Nie można jednak czegoś takiego ciągnąć w nieskończoność i w końcu autor zdecydował się sprostować swoją omyłkę. Na schemacie widocznym na rysunku 2 złącze Z2 zostało już narysowane

poprawnie i tak będzie rysowane w kolejnych projektach serii robotyki i innych.

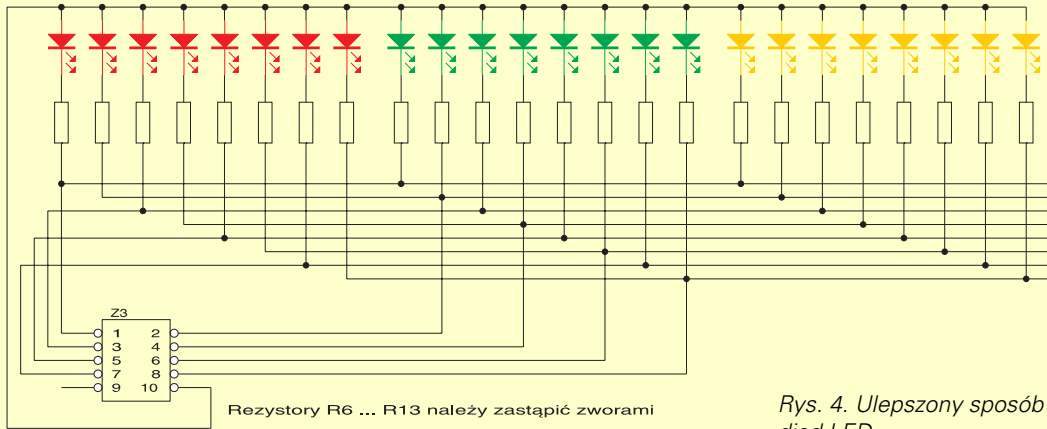
Przejdźmy wreszcie do analizy schematu, którego centralnym punktem jest oczywiście pamięć EPROM. Czytelnikom, którzy nie stosowali dotąd tego ciekawego elementu autor winien jest parę słów objaśnienia. Pamięć typu EPROM (Erasable Programmable Read Only Memory - reprogramowalna pamięć stała) jest najpopularniejszym rodzajem pamięci stałej posiadającej możliwości wielokrotnego programowania. Programowanie pamięci odbywa się za pomocą specjalnego urządzenia - programatora EPROM, współpracującego z komputerem. Ilość informacji jaką możemy zapisać w epromie zależy od typu pamięci. Wykorzystywany w naszym układzie EPROM ma pojemność 8kB, czyli że możemy w nim zapisać 8192 słów 8-bitowych. Jeżeli zapisane w pamięci dane nie są już nam potrzebne, to możemy je łatwo usunąć. Kasowanie pamięci EPROM odbywa się w kolejnym specjalnym urządzeniu - kasowniku EPROM'ów i polega na naświetleniu struktury układu promieniowaniem ultrafioletowym o odpowiedniej długości fali. Być może dla wielu Czytelników nie posiadających

jeszcze komputerów informacja o sposobie programowania pamięci zabrzmiała groźnie. Nie obawiajcie się, w kicie będzie dostarczany zaprogramowany EPROM, a o zapewnieniu Wam możliwości samodzielnego programowania tych pamięci, jeszcze pomyślimy.

Po tej kolejnej dygresji wracamy wreszcie do opisu układu. Cały obszar pamięci EPROM został podzielony na osiem banków pamięci pojemności po 1024B. W każdym z banków zapisane są inne efekty świetlne, a wyboru aktualnego banku dokonujemy za pomocą przełączników S1. Zwarcie jednego z przełączników S1 powoduje podanie na odpowiadające mu wejście adresowe pamięci stanu logicznego 0. W tabeli 1 przedstawiono, w jaki sposób możemy wybrać aktualnie potrzebny bank i uzyskać wymagane w danym momencie efekty. Być może niektórzy Czytelnicy zaniepokoiili się z powodu małych rozmiarów każdego z banków. Mówiąc o pamięciach najczęściej używamy mega i nawet gigabajtów, a tu tylko nieco więcej niż jeden kilobajt! Policzymy więc trochę: jeżeli częstotliwość zegara w naszym układzie będzie wynosić 1Hz, to kolejne sekwencje aktualnego efektu



Rys. 3. Prosty sposób włączenia diod LED.



Rezystory R6 ... R13 należy zastąpić zworami

Rys. 4. Ulepszony sposób dołączenia diod LED.

światelnego wyświetlane będą co jedną sekundę. A zatem wyświetlenie całego efektu znajdującej się w jednym banku będzie trwało nieco ponad... 17 minut! Maksymalną częstotliwością zegara, przy której oko ludzkie będzie w stanie zarejestrować ruch węża wydaje się być ok. 10Hz. Tak więc nawet przy maksymalnej częstotliwości zegara sekwencje zaczną się powtarzać dopiero po ponad 1,5 minuty. To chyba zupełnie wystarczy, prawda?

Po wybraniu właściwego banku pamięci włączamy zasilanie układu (banki pamięci można zmieniać także podczas pracy urządzenia). Generator zegarowy zbudowany z wykorzystaniem nieśmiertelnego układu NE555 - IC4 rozpoczyna pracę i na wejście dwunastobitowego licznika binarnego IC3 doprowadzony zostaje ciąg impulsów prostokątnych. Częstotliwość sygnału zegarowego możemy zmieniać za pomocą P1, dostosowując tempo prezentacji efektów świetlnych do aktualnych potrzeb. Jako P1 zastosowano potencjometr logarytmiczny, ze względu na czysto użytkowe walory regulacji "płynięcia" światła. Potencjometr w wersji liniowej (A) daje intuicyjnie mniejszą dokładność regulacji.

Wyjścia licznika IC3 adresują dziesięć "młodszych" wejść adresowych pamięci EPROM powodując ustawianie na wy-

jściach Q₀...Q₇ kombinacji logicznych odpowiadających kolejnym sekwencjom aktualnie wyświetlanego efektu. Wyjścia pamięci połączone są z wejściami układu IC2 - scalonego drivera średniej mocy. Każde z wyjść tego układu możemy obciążyć prądem o wartości do 500mA, co daje możliwość dołączenia do każdego z kanałów do 20 diod świecących połączonych równolegle (zakładamy, że każda z diod pobiera prąd ok. 20mA).

Pozostała część układu to typowo skonstruowany zasilacz, wykorzystujący scalony monolityczny stabilizator napięcia 7805.

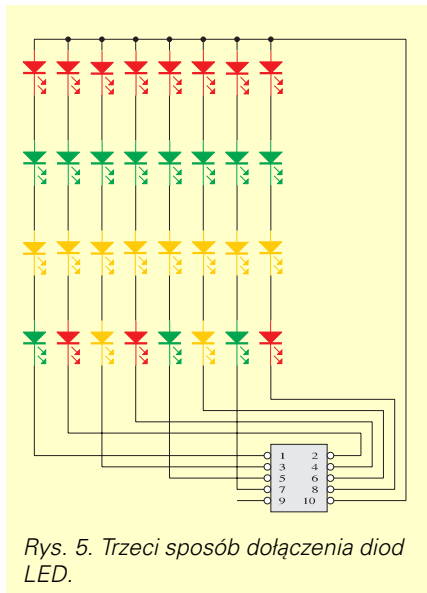
Wiemy już wszystko o działaniu naszego sterownika węża świetlnego i najwyższa pora powiedzieć parę słów o sposobach dołączania do niego elementów wykonawczych. Zajmiemy się najpierw diodami LED, a o możliwości zastosowania innych źródeł światła pomówimy nieco później.

Diody LED możemy dołączyć do układu na trzy sposoby. Pierwszy z nich pokazany jest na **rysunku 3** i od razu widać, jakie ograniczenia ze sobą niesie. Przy równoległym połączeniu diod z wykorzystaniem ośmiu rezystorów wlotowanych w płytkę możemy stosować tylko diody jednakowego koloru. Więcej, zaleca się stosować diody identycznego typu i od tego samego producenta.

Nieco odmienny sposób montażu diod LED przedstawia **rysunek 4**. W tym przypadku wszystkie diody wyposażone zostały w rezystory szeregowo ograniczający płynący przez nie prąd. To rozwiązanie daje nam dość dużą swobodę działania, kolor i typ zastosowanych diod nie ma żadnego znaczenia. Możemy z nich układać zupełnie dowolne kombinacje, wzbogacając w ten sposób uzyskiwane efekty. Ogromną wadą tego sposobu montażu jest konieczność przylutowania do diod ogromnej ilości rezystorów. Ich wartość należy dostosować do napięcia zasilania, tak aby prąd płynący przez diody nie przekraczał 20mA dla każdej z nich. Obydwa omówione sposoby montażu diod mają jedną, wspólną wadę: konieczność stosowania zasilacza o względnie małym napięciu i dużej wydajności prądowej.

Wady tej nie posiada rozwiązanie trzecie, przedstawione na **rysunku 5**. W tym przypadku diody LED zostały połączone szeregowo. Możemy teraz także dowolnie mieszać kolory i typy diod, ale musimy zastosować zasilacz o większym napięciu (ale za to o mniejszej wydajności prądowej). Mamy następujące ograniczenia:

1. Napięcie zasilacza musi być większe od spadku napięcia na wszystkich diodach połączonych w szereg.
2. Napięcie zasilania nie może przekro-



Rys. 5. Trzeci sposób dołączenia diod LED.

czyć 40V. Przekroczenie tego napięcia groziłoby uszkodzeniem układu ULN2803.

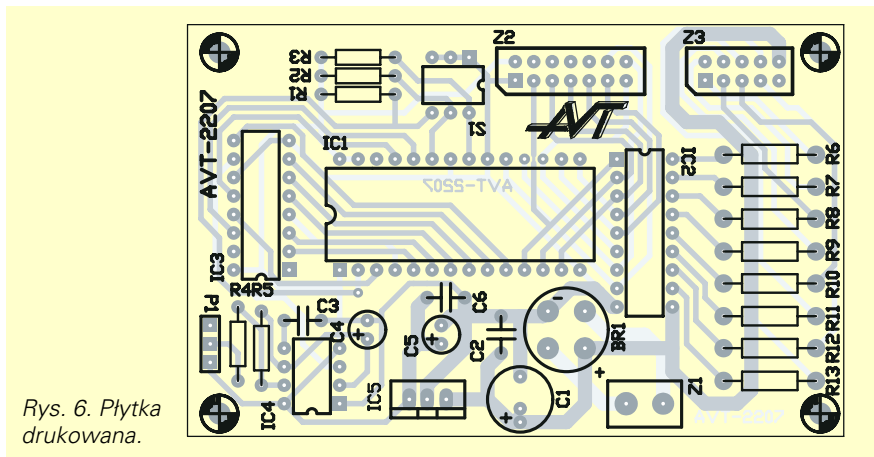
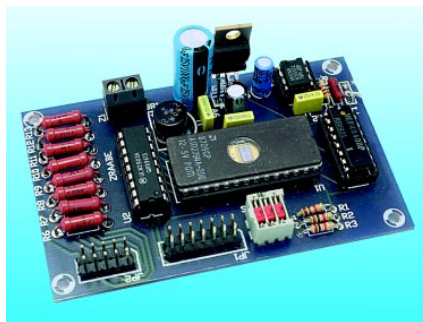
Montaż i uruchomienie

Rozmieszczenie elementów pokazano na rysunku 6, a mozaikę ścieżek płytki drukowanej na wkładce. Montaż wykonujemy w tradycyjny sposób, rozpoczynając od najmniejszych elementów, a kończąc na kondensatorach elektrolitycznych i stabilizatorze napięcia. Autor zawsze zaleca stosowanie podstawek pod układy scalone, a wlotowanie podstawki pod pamięć EPROM jest absolutnie konieczne. Możemy przecież zaprogramować sobie w przyszłości własne EPROMy, zawierające nowe efekty świetlne i musimy mieć możliwość wymiany pamięci.

Zmontowany starannie i ze sprawdzonych elementów układ nie wymaga żadnego uruchamiania ani regulacji. Już podczas jego pracy możemy jedynie dobrać stosownie do naszych potrzeb częstotliwość generatora zegarowego, a tym samym szybkość "poruszania się" węża.

Autor wspominał w pierwszej części artykułu o możliwości współpracy naszego sterownika z innymi układami serii 2000. Rzeczywiście, taka możliwość istnieje, możemy wykorzystać aż trzy dodatkowe układy wykonawcze:

1. AVT-2099 Moduł wykonawczy średniej mocy. Możemy go dołączyć równolegle do układu, wykorzystując specjalnie do tego przeznaczone złącze Z2. Moduł ten zawiera w sobie układ scalonego drivera mocy typu ULN2803, podobnie jak nasz sterownik. Stosując moduł (lub większą ilość tych modułów) możemy wielokrotnie zwiększyć ilość dołączonych do sterownika diod, lub innych obciążeń (np. żarówczek choinkowych).
2. AVT-2098 Moduł wykonawczy dużej mocy. Tu możliwości mamy naprawdę ogromne: do naszego sterownika możemy dołączyć obciążenie do 8A na każdy kanał i mogą to być nawet żarówki zasilane z sieci 220VAC. Takim wężem można już oświetlić spore pomieszczenie! Jest tu jednak jeden haczyk: moduł AVT-2098 skonstruowany jest na przekaźnikach i podczas działania wydaje nieprzyjemne dla ucha trzaski. Przy dużej częstotliwości pracy sterownika węża hałas byłby prawdopodobnie trudny do zniesienia. Pozostaje więc umieszczenie modułu wykonawczego w miejscu, w którym hałas nikomu nie przeszkadza, albo zastosowanie.



Rys. 6. Płytki drukowana.

3. AVT-2097 Moduł wykonawczy dużej mocy na triakach. Układ ten jest obecnie testowany i jego opis zamieścimy w jednym z najbliższych numerów EdW. Nadaje się on do naszych celów wprost idealnie: umożliwia dołączenie do sterownika węża żarówek o mocy ograniczonej jedynie dopuszczalnym prądem zastosowanych triaków, a ponadto nie wydaje żadnych dźwięków i nie generuje zakłóceń radioelektrycznych.

Program sterujący

Pozostała nam obecnie jeszcze jedna sprawa do omówienia. Z pewnością wielu Czytelników posiada komputery i programatory EPROM i chcieliby wykonać własne EPROMy do swojego sterownika węża. Tym Czytelnikom autor pragnie polecić własną, sprawdzoną metodę obliczania liczb, które należy umieścić w kolejnych komórkach pamięci. Do przygotowania takiego programu potrzebny jest dowolny komputer (nawet Commodore-C64 z systemem operacyjnym GEOS) i arkusz kalkulacyjny. Autor posługiwał się komputerem klasy PC i arkuszem MS Excell, ale można zastosować dowolny inny arkusz kalkulacyjny, np. Lotus. Kolejność postępowania jest następująca:

W pierwszej kolumnie arkusza wpisujemy formułę przeliczającą zapis binarny na dziesiętny, tak jak pokazano na rysunku 7. Komórkę z formułą musimy na-

WYKAZ ELEMENTÓW

Rezystory

- P1: potencjometr 100kW/B
- R1...R3, R5: 1kW
- R4: 10kW
- R6...R13: 68W/0,5W

Kondensatory

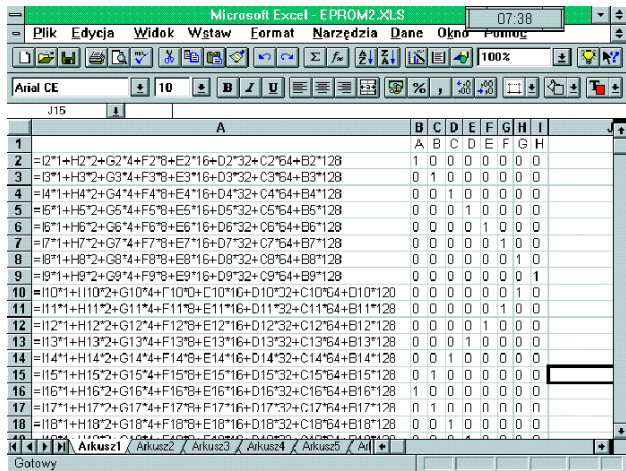
- C1: 1000µF/25V
- C2, C6: 100nF
- C3: 10nF
- C4: 3,3µF/16V
- C5: 100µF/6,3V

Półprzewodniki

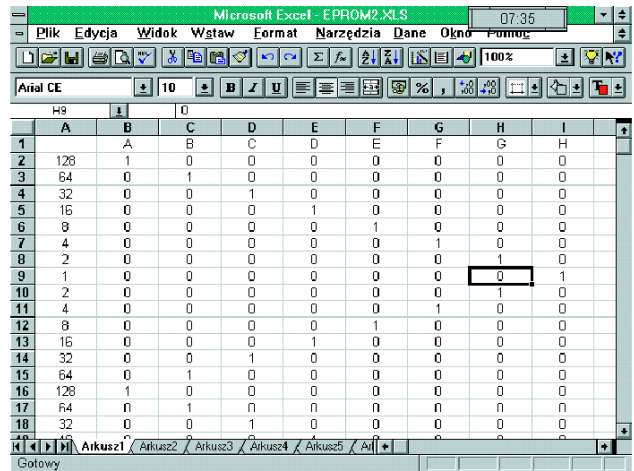
- BR1: mostek prostowniczy 1A
- IC1: zaprogramowana pamięć EPROM 2764
- IC2: ULN2803
- IC3: 4040
- IC4: NE555
- IC5: 7805

Różne

- S1: SW DIP-3
- Z1: ARK2
- Z2: goldpin 2x14
- Z3: goldpin 2x5



Rys. 7.



Rys. 8.

stępnie przekopiować w dół do kolejnych 8191 komórek.

Następnie cały obszar roboczy: kolumny B, C, D, E, F, G, H i I zapelniamy zerami. Przy odrobinie wprawy w posługiwaniu się arkuszem kalkulacyjnym opisane czynności nie zajmą nam więcej niż minutę.

W kolumnach obszaru roboczego 1 reprezentuje zapaloną diodę, a 0 zgaszoną. Chyba teraz każdy może ocenić, jak wygodna jest proponowana metoda tworzenia programu do EPROMu: po prostu graficznie przedstawiamy w arkuszu to, co zostanie wyświetlone przez sterownik! Na **rysunku 8** podano najprostszy przykład: przesunięcie zapalnego punktu w prawo, a potem w lewo oraz wyniki konwersji kodu binarnego na dziesiętny w kolumnie A. Oczywiście, podczas tworzenia programu można, a nawet należy posługiwać się metodą kopiowanie bloków obszaru roboczego.

Pozostaje już tylko zapisać liczby znajdujące się w kolumnie A w postaci pliku ASCII, a następnie dokonać konwersji tego pliku do postaci binarnej, czytelnej dla programatora EPROM.

Modyfikacje

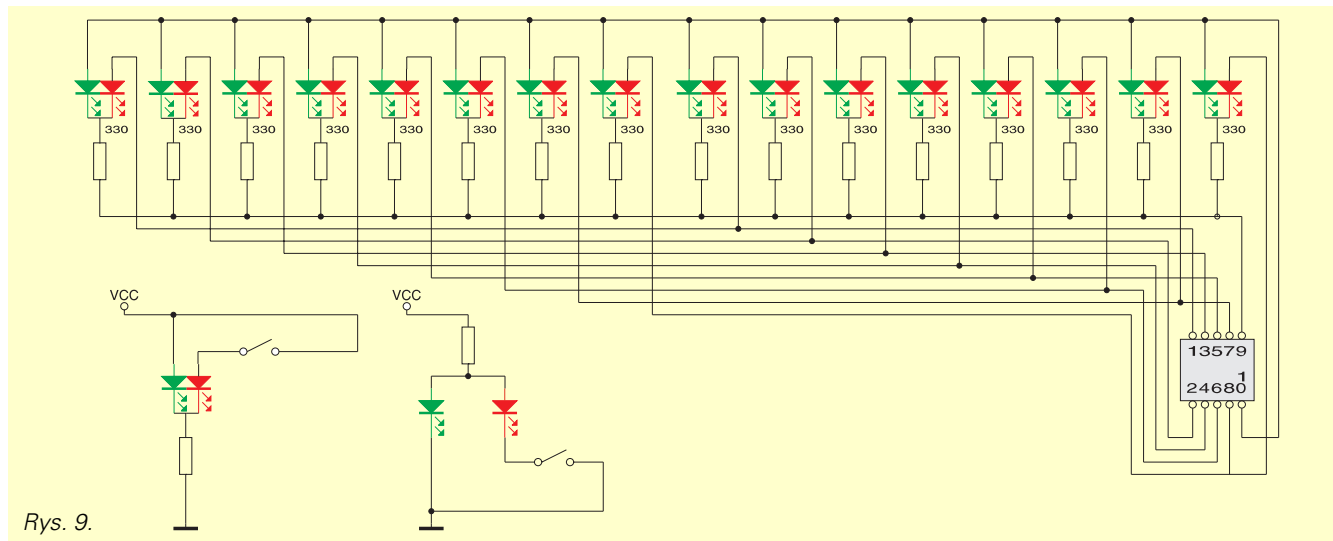
Istnieje jeszcze jeden sposób wykonania węża świetlnego. Może nie wszyscy Czytelnicy wiedzą, że istnieje pewien szczególny rodzaj diod LED: diody dwubarwne. Posiadają one trzy wyprowadzenia: wspólną katodę i dwie, oddzielone od siebie anody. Najbardziej popularne są diody dwubarwne czerwono - zielone. W najbliższej przyszłości zajmiemy się efektami świetlnymi, które możemy uzyskać za pomocą tych ciekawych elementów, a teraz pomyślny jak dołączyć je do naszego sterownika. Wykorzystamy tu fakt, że napięcia odkładające się na diodach LED o różnych kolorach są także różne. Napięcie na diodzie czerwonej wynosi ok. 1,9V, natomiast na diodzie zielonej ok.2,2V. Co zatem się stanie, jeżeli połączymy ze sobą anody diody dwubarwnej i zasilimy je wspólnie poprzez jeden rezystor? To oczywiste: zapali się tylko dioda czerwona, zwierając diodę zieloną. Jeżeli natomiast odłączymy anodę diody czerwonej, to zapali się dioda zielona.

Na **rysunku 9** widoczne są dwa przykłady łączenia równoległego diod LED o dwóch kolorach i schemat dołączenia do naszego sterownika girland składających się z dwubarwnych LED. Efekt może być interesujący: wszystkie zapisane w EPROM efekty będą teraz wyświetlane na zielonym "tle".

Tak zmontowany układ nie będzie jednak działał bez kilku przeróbek:

- rezystory R6 i R13 należy zastąpić zworkami.
- zamiast układu ULN2803 należy włożyć w podstawkę układ typu TD62768 lub jego odpowiednik. TD62768 jest driverem o parametrach podobnych do ULN2803 z jedną istotną różnicą: zasila on odbiorniki prądu od strony plusa, a nie minusa zasilania.
- należy przeciąć ścieżkę prowadzącą do nóżki 9 tego układu i doprowadzić do niej dodatnie napięcie zasilania sprzed stabilizatora napięcia.

Zbigniew Raabe



Rys. 9.