

Wyświetlacz 3D

Zaskakujące trójwymiarowe ruchome motywy i efekty cieszą oko każdego widza. Chcesz wiedzieć, jak je stworzyć? Zapoznaj się z artykułem i przekonaj, że to wcale nie takie trudne, jak może się wydawać.

W EdW były już przedstawiane różne projekty wyświetlaczy, jednak wersji trójwymiarowej jeszcze nie było! Być może jako miłośnik elektroniki już spotkałeś się z podobnymi projektami wyświetlaczy, chociażby na internetowych portalach video.

Prawdopodobnie uważasz, że budowa takiego przestrzennego wyświetlacza to coś bardzo trudnego. Okazuje się, że prawda jest inna.

Owszem, trzeba troszkę znać się na programowaniu. Owszem, budowa samej przestrzennej konstrukcji wymaga trochę cierpliwości. Jednak w sumie uzyskane efekty z nawiązką rekompensują trud włożony w wykonanie. Cały układ składa się z powszechnie dostępnych i tanich elementów.

Ciekawostką jest fakt, że płytką do pierwszego autorskiego modelu została zaprojektowana zupełnie nietypowo – rysunek ścieżek powstał w programie do rysowania... Paint.

Prezentowane urządzenie nie ma konkretnie ukierunkowanego przeznaczenia. Daje duże pole do popisu posiadaczowi. Może służyć na przykład jako lampka nocna w pokoju dziecięcym, ozdoba na biurko, a także jako wyszukana reklama. Wyświetlacz może również świetnie sprawdzić się w roli przyrządu do ćwiczeń programowania. Także jeśli ktoś rozpoczyna dopiero przygodę z elektroniką – to urządzenie z pewnością go zainteresuje, bo jego realizacja naprawdę nie jest tak trudna, jak mogłoby się wydawać.

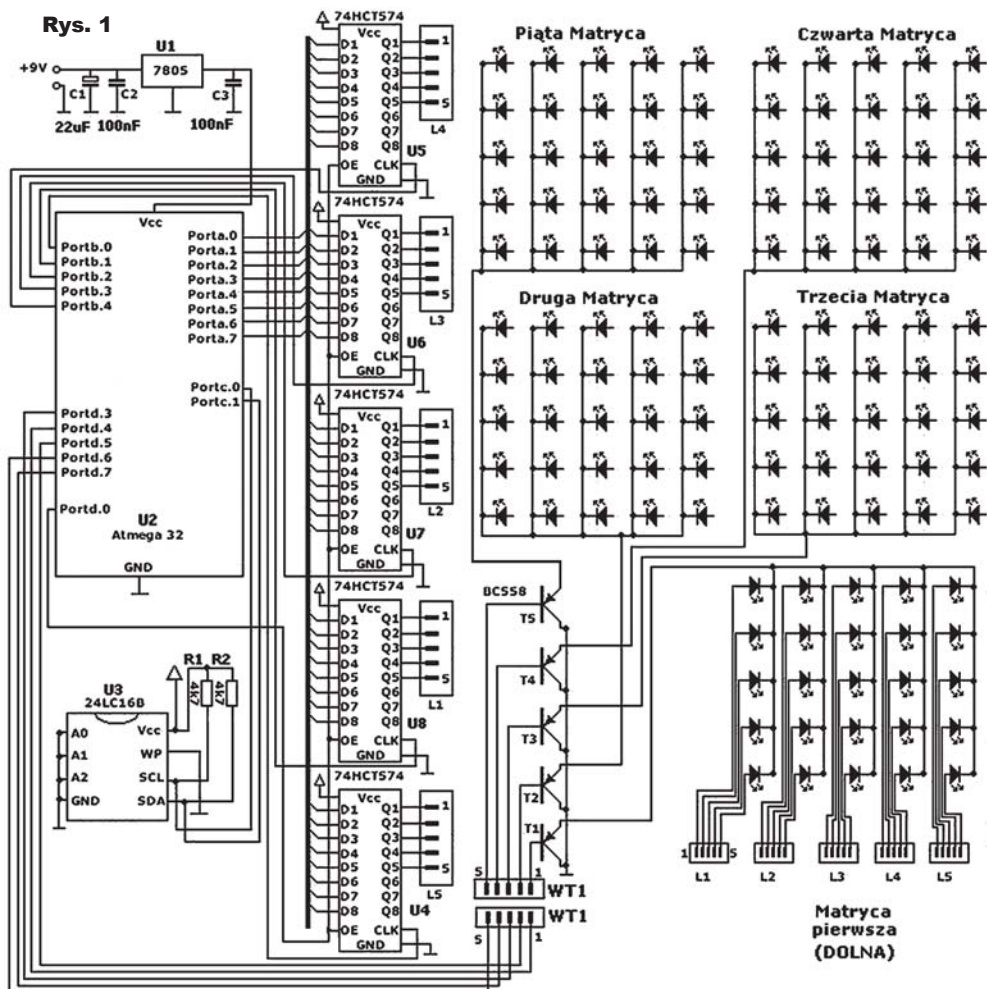
Warto zwrócić uwagę na niekończące się możliwości tego układu. Liczba możliwych do wyświetlenia motywów i figur jest praktycznie nieskończona – wszystko zależy od kreatywności osoby programującej.

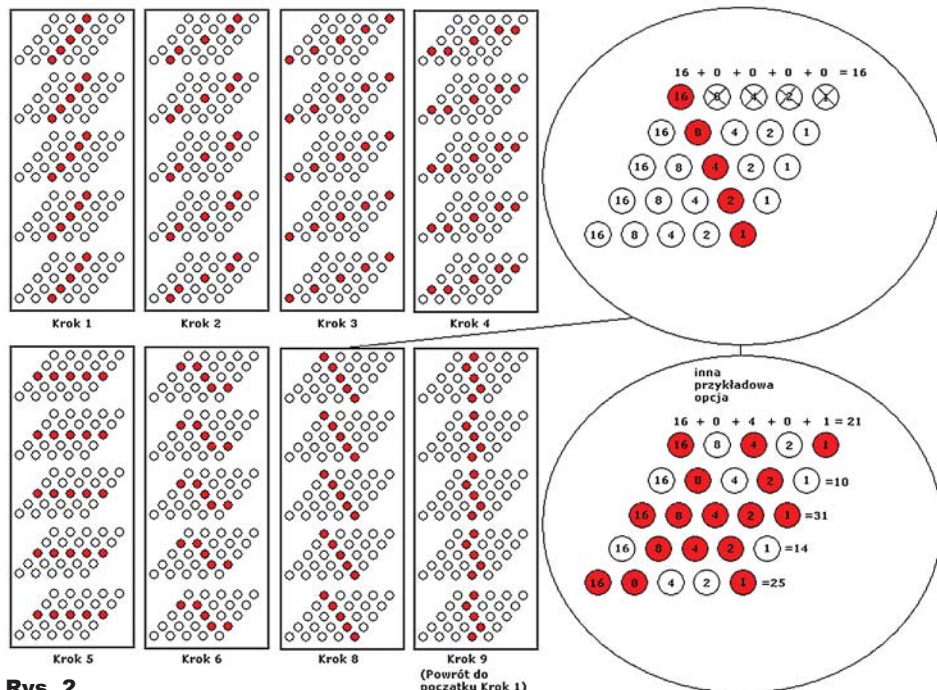
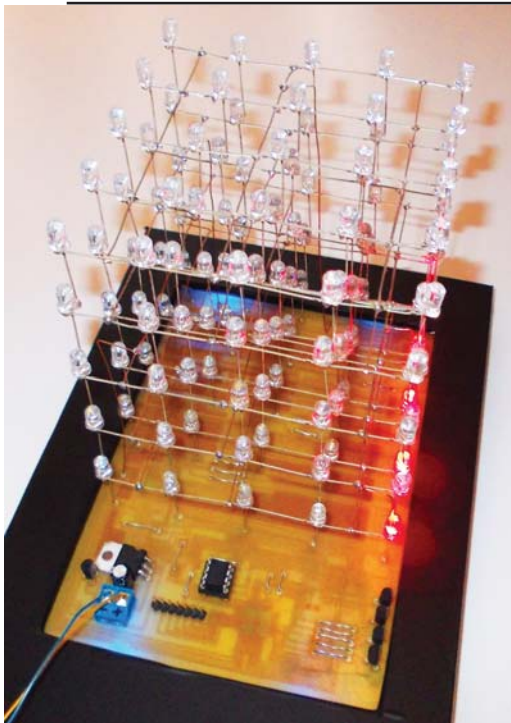
Opis układu

Schemat wyświetlacza pokazany jest na **rysunku 1**. Układ sterowany jest popularnym mikrokontrolerem ATmega32. PortA podłączony jest równolegle do wszystkich wejść zatrząsków, rejestrów 8-bitowych (74HCT574). Chociaż do wysterowania całego wyświetlacza potrzebnych jest tylko 25 wyjść, do



Rys. 1





Rys. 2

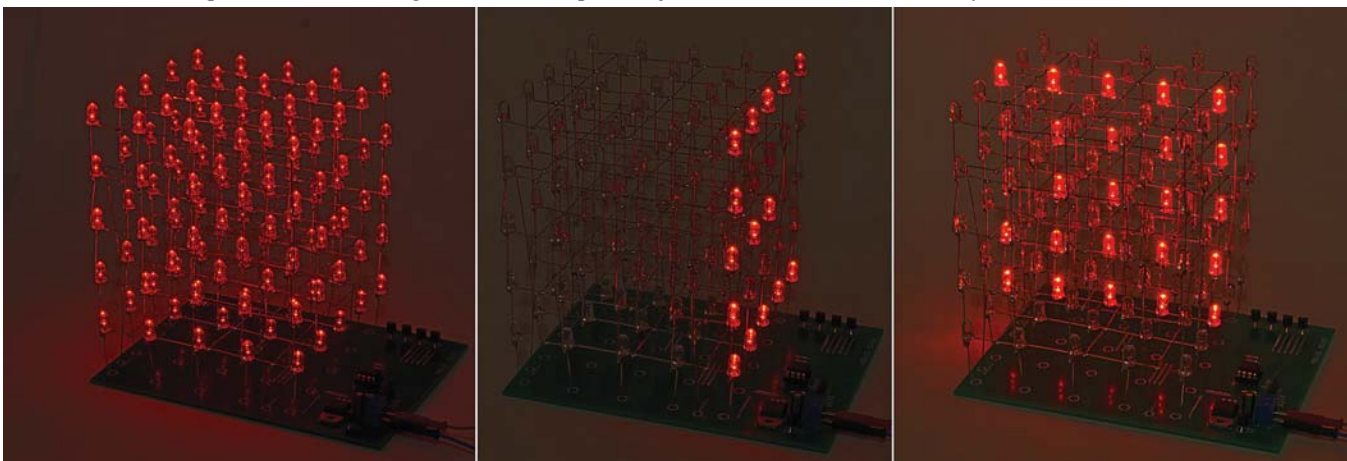
których w zupełności wystarczyłyby 4 rejestry, ja użyłem jednego więcej, by ułatwić sobie później programowanie. Portd.0 podpięty jest do wejścia rejestru oznaczonego OE, czyli do złącza udostępniającego sterowanie wyjściami tego rejestru. Z kolei wszystkie wyjścia CLK sterowane są za pomocą Portów b.0–b.4 i dzięki nim możemy wczytać do każdego z rejestrów określoną liczbę w postaci binarnej, dzięki czemu zostanie wyświetlony obraz na wyświetlaczu LED. Porty d.3–d.7 podłączone są do baz tranzystorów BC558, które warunkują zasilanie każdej z matryc, poczynając od dolnej, kończąc na górnej. Jak widać na schemacie ideowym, zastosowana również została pamięć EEPROM, do której podłączone zostały dwa rezystory R1 i R2, niezbędne do jej prawidłowego działania. Linie pamięci SDA i SCL podpięte są do Portów c.0 i c.1. Pamięć EEPROM w tym projekcie nie została wykorzystana – potraktowałem ją raczej jako moduł na przyszłość, rozszerzający możliwości tego urządzenia. Zasilanie układu zapewnia stabilizator napięcia 7805, którego

maksymalny prąd obciążenia w zupełności wystarczy, ponieważ dzięki nowoczesnym diodom LED urządzenie ma bardzo niski pobór prądu.

Kluczową rolę w tym projekcie odgrywa zawarty w mikrokontrolerze program, który został napisany w języku Bascom AVR. Trudno byłoby opisać działanie tego wyświetlacza, nie wspominając o najważniejszym – o tym, co nim steruje i w jaki sposób powstają wyświetlane animacje. Otóż w momencie początkowym procesor udostępnia rejestry, podając im stan niski na złącza oznaczone OE (output enabled). W tym samym czasie zostaje zasilona matryca pierwsza (dolna) poprzez podanie stanu niskiego (0/L) na bazę tranzystora T1. Figury ukazujące się na wyświetlaczu, w programie zapisane są za pomocą liczb w systemie dziesiętnym. Podczas pracy programu zostają przekształcone przez mikrokontroler na zapis w systemie dwójkowym, czyli binarnym. Tak więc rozpoczyna się przekaz pierwszej wartości binarnej do pierwszego rejestru (U4). Następnie na złącze zegarowe tego rejestru podawany jest stan wysoki, a zaraz potem niski, co powoduje, że wartość, która

pojawia się na jego wejściu, zostaje przetrzucona na jego wyjściu. Kolejna, nowa wartość zostaje odczytana w mikrokontrolerze i skierowana do następnego rejestru, wykonując po drodze te same operacje co w pierwszym przypadku. I tym sposobem każdy z pięciu rejestrów otrzymuje przekazaną mu informację. Wartości te wędrują do wszystkich matryc, jednak pokażą się jedynie na pierwszej dolnej, z racji tego, że w tym momencie tylko ta jest zasilona. Na bazę tranzystora T1 zostaje wysłany stan wysoki, co odcina zasilanie z pierwszej matrycy, a do rejestrów podana zostaje wartość 0. Zaraz po tym nowe wartości binarne zostają wpisane do rejestrów, a stan niski na bazie tranzystora T2 umożliwia ukazanie się ich na matrycy drugiej. I tak analogicznie aż do ostatniej matrycy, czyli piątej (górnej), co w całości tworzy trójwymiarowy obraz. Jednakże obraz ten jest zaledwie pojedynczą klatką rozbudowanych, „poruszających się” slajdów.

Na **rysunku 2** został przedstawiony sposób, w jaki można przykładowo projektować obraz na jednej z matryc. Wszystkie wartości dziesiętne zawar-

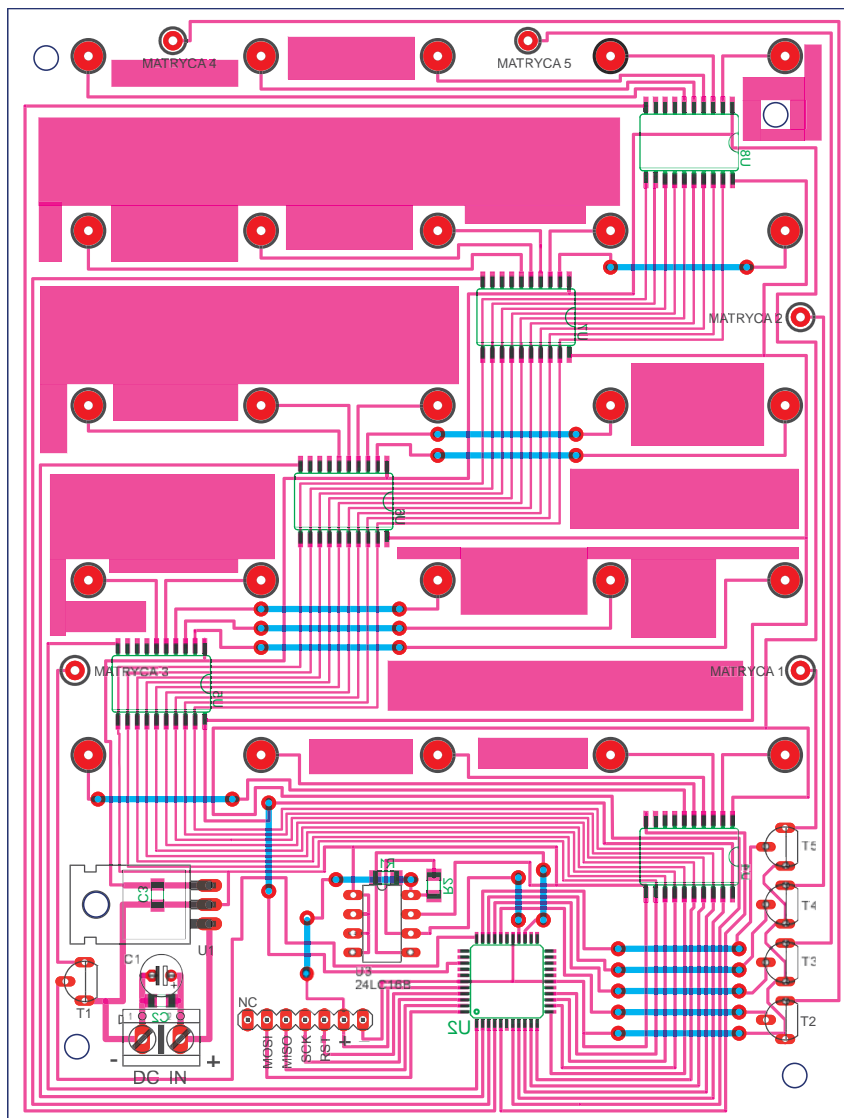


te są w liniach „Data” zamieszczonych w programie mojego autorstwa. Każda pierwsza wartość stałej poszczególnych etykiet tworzy na wyświetlaczu jeden slajd, czyli składową trójwymiarowego motywu. Kolejna, druga wartość odczytana przez program to następny slajd i tak do wyczerpania wpisanych wartości stałych. Potem koło się zapętla i program odczytuje kody od nowa. Wszystkie pliki programu (BASCOM AVR) wraz z jego opisem dostępne są w Elportalu, wśród materiałów dodatkowych do tego zadania.

Montaż i uruchomienie

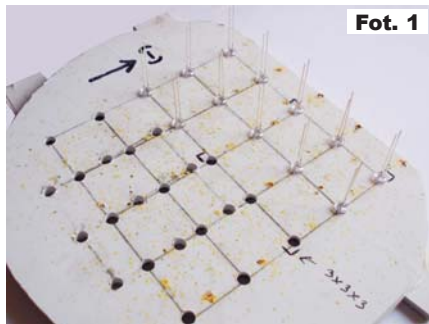
Układ można zmontować na płytce drukowanej, której projekt pokazany jest na rysunku 3.

Standardowo montujemy układ, zaczynając od elementów najmniejszych, a kończąc na największych. Ponieważ część podzespołów jest elementami SMD, proponuję zacząć właśnie od tej trudniejszej strony. W pierwszej kolejności lutujemy najmniejsze elementy, czyli rezystory R1 i R2 oraz kondensatory C2 i C3. Następnie układy scalone 74HCT574, zaczynając od pierwszego (U4), kończąc na ostatnim piątym (U8). Na sam koniec lutujemy mikrokontroler ATmega32. Przy montażu tego procesora należy zwrócić szczególną uwagę na jego polaryzację, która powinna zgadzać się z tą oznaczoną na płycie. Mamy wtedy gotową stronę dolną – *bottom*. Teraz zajmiemy się montażem przewlekającym, czyli *trough hole*. Zaczynamy od wlotowania wszystkich zwór na stronie elementów *top*. Następnie lutujemy elementy podstawki precyzyjnej pod pamięć EEPROM 24LC16B (U3). Potem montujemy stabilizator napięcia 7805 (U1), tranzystory BC558 (T1–T5), kondensator (C1) o pojemności 22uF oraz złącza DC i goldpin. Montaż elementów SMD płyty mógł przysporzyć nieco trudności, jednak

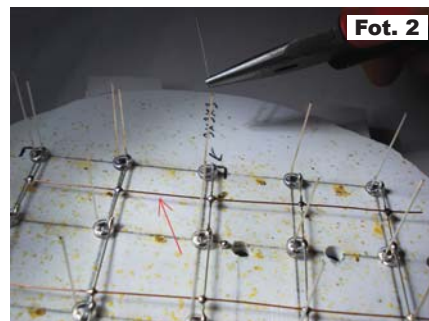


Rys. 3 w porównaniu do stworzenia sześcianu LED jest on dziecinnie prosty. Mając polutowaną płytkę, przechodzimy do najtrudniejszego etapu, czyli złożenia 125 LED-ów w jedną spójną całość.

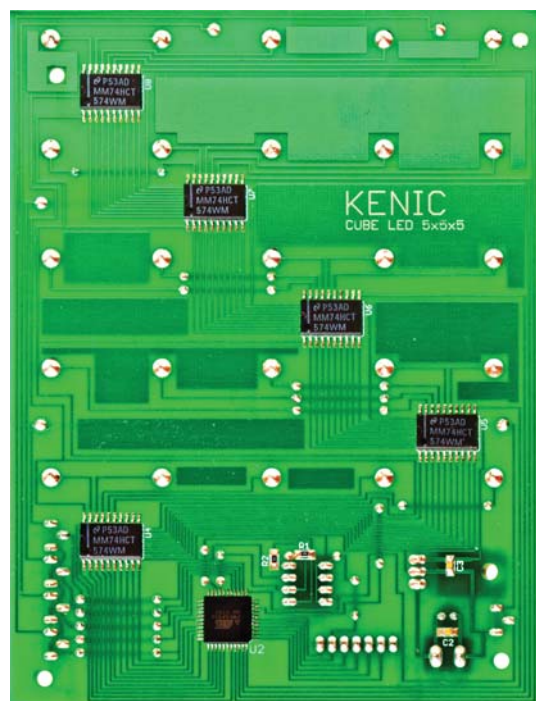
Do równego zmontowania wszystkich diod będzie nam potrzebny jakiś sensowny sposób. Tym sposobem jest szablon (**fotografia 1**), który musimy wykonać sami. Może on być wykonany z jakiegokolwiek płyty, w której da się wywiercić otwory o średnicy 5mm, do których wkładać będziemy później nasze diody LED. Ja wybrałem płytę PCV o grubości 5mm, w której średnio co 2,5cm wywiercałem otwory. Ważne, aby odległości nie były większe niż długość nóżki diody (katody). Na takim szablonie można jednorazowo wykonać jeden poziom (jedną matrycę), gdzie 25 LED-ów ułożonych jest w równych odległościach w kształcie wypełnionego kwadratu. Wszystkie



Fot. 1



Fot. 2



diody w szablonie zwrócone są katodami w jednym kierunku. Łączymy je ze sobą w pięciu poziomych rzędach. Następnie wszystkie te rzędy trzeba połączyć w całość. Ja użyłem do tego miedzianego drutu, który wyciągnąłem z kabla 6-żyłowego używanego do instalacji alarmowych. Drut o grubości 0,5mm uzyskałem, obierając z izolacji jedną wyciągniętą żyłę z przewodu. Tak więc łączymy katody diod LED w trzech rzędach (**fotografia 2**). Po wyciągnięciu z szablonu taka matryca powinna być w miarę stabilna, jednak trzeba uważać, gdyż jest delikatna i bardzo łatwo ją powyginać. Tak wykonaną matrycę odkładamy i przystępujemy do montażu następnej, takiej samej, aż uzyskamy 5 gotowych poziomów.

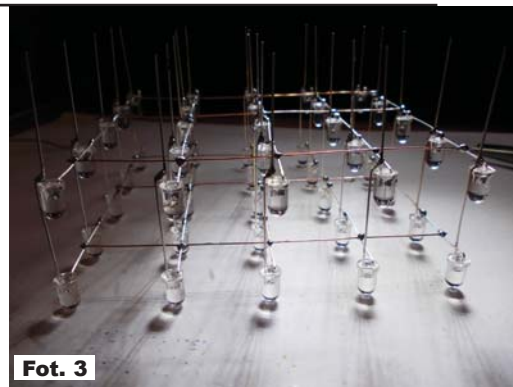
Następnym etapem jest łączenie ze sobą dwóch pierwszych matryc. Naszym zadaniem będzie połączenie wszystkich anod obu matryc w pionie. Aby ułatwić sobie zadanie, zacząłem montaż od zlutowania czterech rogów kwadratu, dzięki czemu oba poziomy będą wystarczająco stabilne, by wykonać pozostałych 21 połączeń. Na zdjęciu widać złączone ze sobą dwie matryce (**fotografia 3**). Przystępujemy do montażu kolejnej, gdzie należy pamiętać, żeby przy lutowaniu diody były ustawione „do góry nogami”, czyli elektrodami w górę, co znacząco ułatwi dostęp grotu lutownicy. Trzeba pamiętać również o tym, aby *po każdym połączeniu matrycy starannie sprawdzić jakość lutowania i w razie wypadku poprawić* (pod lupą czy, jeśli ktoś posiada, pod mikroskopem lub stereoskopem). Piszę o tym, ponieważ po zmontowaniu całego sześcianu nie będzie możliwości poprawy lutowania jednego z połączeń np. w samym centrum struktury, gdyż zwyczajnie dostęp będzie mocno ograniczony i próby takiej naprawy mogłyby przysporzyć jeszcze więcej problemów.

Kiedy sześcian będzie już gotowy, a jego połączenia dokładnie sprawdzone, nie pozostaje nam nic innego, jak połączenie modułu sterującego (płyty PCB) z trójwymiarowym wyświetlaczem. Aby tego dokonać, pierwszą rzeczą, którą się zajmujemy, będzie wyprostowanie anod z dolnej matrycy, dzięki czemu ich montaż w wywierconych otworach na płycie będzie łatwiejszy. Aby ułatwić sobie przewlekanie nóżek przez otwórki w płycie,

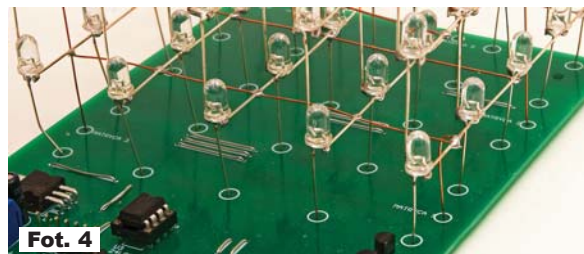
najlepiej użyć do tego pincety. Kiedy ten czasochłonny proces będziemy mieli za sobą, należy odwrócić płytę na stronę *bottom*, czyli położyć wyświetlacz na diodach, i zlutować pady, przez które wystają nóżki diod LED. Mała uwaga: przy lutowaniu najlepiej nie używać topników, a korzystać z cyny, w której wnętrzu jest sama kalafonia. Płynny topnik może pod wpływem grawitacji przedostać się przez otwór na drugą stronę płyty (top) i ją pobrudzić. Usunięcie go z tak niedostępnego miejsca byłoby niepotrzebnym problemem.

Tak więc mamy już prawie gotowy wyświetlacz LED. Całość trzyma się mocno na płycie i mogłoby się wydawać, że wszystko jest już gotowe do użycia. Na koniec jednak został nam jeszcze jeden mały szczegół. Trzeba ponownie użyć miedzianych drutów, aby dolutować do każdej z matryc sterowanie jej zasilaniem. Przy trzech zewnętrznych bokach sześcianu na płycie mieszczą się otwory, przez które podłączymy sterowanie zasilaniem każdej z matryc. Wszystkie otwory są opisane i dedykowane konkretnemu poziomowi (**fotografia 4**).

Po wykonaniu tych połączeń wyświetlacz nareszcie jest gotowy i nie pozostaje nam nic innego jak podłączenie zasilania i uruchomienie go. Będziemy do tego potrzebowali zasilacza o natężeniu



Fot. 3



Fot. 4

minimum 500mA i napięciu 12V. Całość wymaga oczywiście także wprowadzenia do procesora programu z trójwymiarowymi motywami. Filmik pokazujący prace modelu (w dwóch rozdzielczościach), pliki z projektem i rysunkami płytki oraz przykładowy program można znaleźć w Elportalu, wśród materiałów dodatkowych do tego numeru.

Marcin Strehlau
marcin-strehlau1@wp.pl

Wykaz elementów

Rezystory:

R1 4,7kΩ
R2 4,7kΩ

Kondensatory:

C1 22μF elektrolityczny
C2 100nF ceramiczny
C3 100nF ceramiczny

Półprzewodniki:

D1–D125 diody LED czerwone przezroczyste
U1 7805
U2 ATmega32-AU (TQFP44)
U3 24LC16B
U4–U8 74HCT574 (SMD)
T1–T5 BC558

Inne:

Gniazdo DC
Wtyk goldpin (7 pinów)
Podstawka precyzyjna (DIP8)

Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-3060.

