

wyświetlacz ze wspólną katodą! Do podstawki dołączamy zasilanie 5VDC (nóżka 20 +, nóżka 10 -) i łączymy ją z emulatorem. Następnie piszemy w edytorze BASCOM-a króciutki programik:

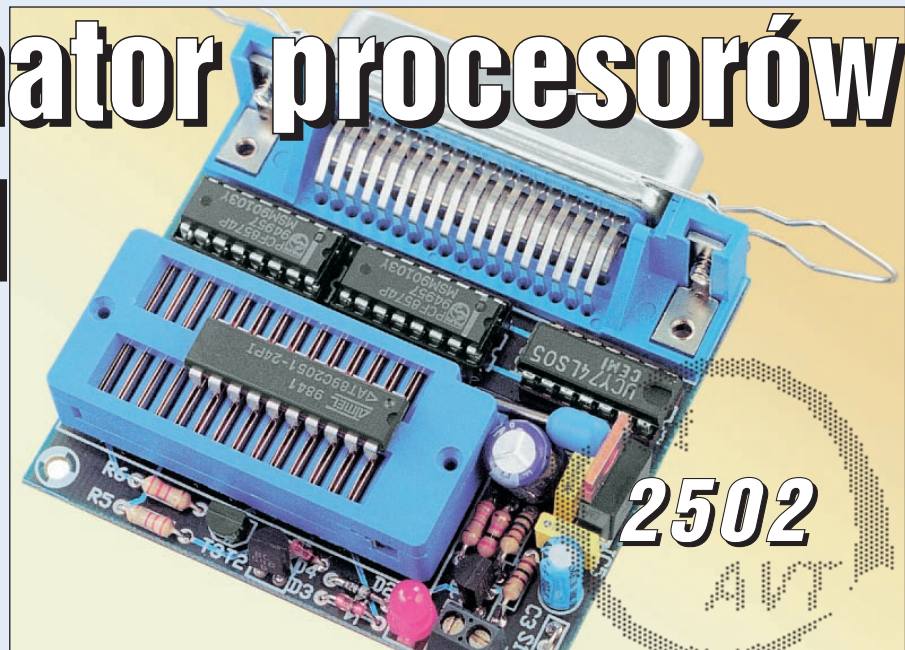
Po jego skompilowaniu uruchamiamy emulator (rys.4) i obserwujemy zachowanie wyświetlacza. Jeżeli zaczną się na nim pojawiać cyfry od 0 do 9 wyświetlane w pętli, to nasz układ emulatora możemy uznać za sprawny i gotowy do testowania napisanych programów.

Programator procesorów 89CX051

Jak już zapewne wiecie, sam program BASCOM to za mało: potrzebne są jeszcze proste narzędzia hardware'owe umożliwiające mu kontakt z procesorem. Jednym z tych narzędzi jest opisany emulator sprzętowy, a drugim, którym zajmiemy się teraz – programator, bez którego wykonywanie układów z mikroprocesorami jest absolutnie niemożliwe.

Programator MCS Flash Programmer został opracowany przez firmę MCS Electronics, która w swej uprzejmości udzieliła mi zgody nie tylko na publikację schematu programatora, ale także na wykonanie płytki obwodu drukowanego, przetestowanie układu i następnie rozprowadzanie go pod postacią kitu.

Programator firmy MCS Electronics jest urządzeniem wręcz uderzającym swą prostotą. Zbudowany został z wykorzystaniem zaledwie trzech tanich i bardzo łatwo dostępnych układów scalonych i garstki elementów dyskretnych. Jego wykonanie jest absolutnie możliwe nawet dla początkujących elektroników, a układ nie wymaga jakiegokolwiek regulacji ani uruchamiania. Jednak "coś za coś", prostota i taniość układu została okupiona dwoma, niezbyt zresztą dokuczliwymi ograniczeniami:



1. Proponowany programator może pracować wyłącznie z programem BASCOM LT lub BASCOM8051.

2. Za pomocą opisanego niżej urządzenia możemy programować wyłącznie procesory serii 89CX051, czyli 89C1051, 89C2051 i 89C4051.

Jeżeli jednak komuś z Was nie odpowiada koncepcja budowy programatora o nieco ograniczonych możliwościach, to w ofercie AVT znajduje się uniwersalny programator obsługujący całą rodzinę '51, zaprojektowany przez Sławomira Surowińskiego. Jest to kit AVT-320, idealne narzędzie dla bardziej zaawansowanych konstruktorów.

Jak to działa?

Schemat elektryczny proponowanego układu został pokazany na **rysunku 1**. Ponieważ układ jest banalnie prosty i opisanie go nie zajmie wiele miejsca, możemy poświęcić chwilę na sprecyzowanie, na czym właściwie polega programowanie procesorów i jaki jest jego algorytm w przypadku kostek rodziny 89CX051.

Zapewne wszyscy wiecie, że mikroprocesor sam jako taki nie "potrafi" dosłownie nic. Aby go "ożywić" i zmusić do wykonywania niekiedy bardzo skomplikowanych czynności, potrzebny jest napisany przez człowieka program, który musi zostać umieszczony w miejscu łatwo "dostępnym" dla procesora.

Każdy mikroprocesor musi dysponować co najmniej dwoma rodzajami pamięci: pamięcią programu i pamięcią danych. Jako pamięć danych wykorzystuje się zwykle pamięć RAM lub/i EEPROM, umieszczone na zewnątrz lub wewnątrz procesora, natomiast pamięcią programu jest z zasady pamięć stała, obecnie najczęściej reprogramowalna (przynajmniej w procesorach stosowanych w konstrukcjach amatorskich). Dawniej jako repro-

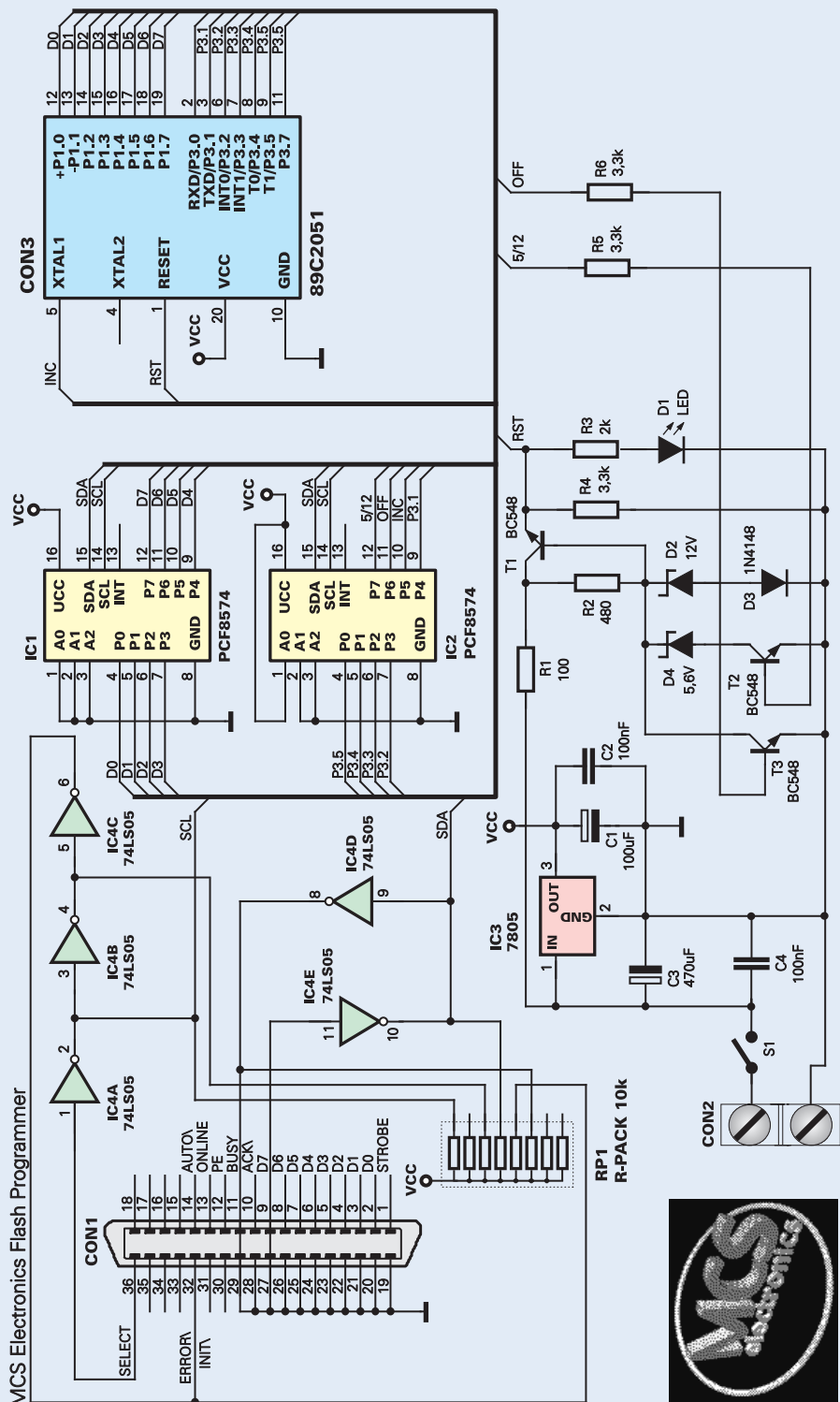
gramowalną pamięć programu stosowano zwykle wewnętrzne lub zewnętrzne pamięci EPROM, obecnie coraz większe uznanie uzyskują pamięci EEPROM, których kasowanie nie wymaga stosowania specjalnych urządzeń i może odbywać się w każdym przystosowanym do tego celu programatorze.

Zalety procesorów z pamięcią programu typu EEPROM szczególnie powinni docenić amatorzy. Taki procesor możemy przeprogramować

praktycznie w nieskończoność, poprawiać napisany program i wpisywać do jego pamięci nowy. Do tego celu potrzebny nam jest jedynie programator, choćby tak banalnie prosty, jak opisywany w tym artykule.

Jakie czynności właściwie wykonuje programator? Przyjrzyjmy się bliżej algorytmowi programowania procesorów serii 89CX051. Pomocą będzie też rysunek 2. Po umieszczeniu procesora przeznaczanego do zaprogramowania w podstawce programatora muszą zostać wykonane następujące czynności:

1. Wymuszenie stanu niskiego na wejściu RST na okres nie krótszy niż 10ms.
2. Wymuszenie stanu wysokiego na wejściach RST i P3.2.
3. Ustawienie odpowiedniej kombinacji stanów logicznych, właściwych dla funkcji, która ma zostać wykonana na wejściach P3.3, P3.4, P3.5 i P3.7. W tabeli poniżej zestawione zostały wszystkie tryby pracy procesora podczas programowania i odpowiadające im stany logiczne na wejściach portu P3.
4. Programowanie i weryfikacja zapisanych danych. Na wejściach portu P1 musi zostać ustawiona kombinacja logiczna odpowiadająca pierwszemu bajtowi wpisywanego do pamięci programu (adres 000H).
5. Dołączenie do wejścia RST napięcia dokładnie równego +12VDC.
6. Podanie na wejście P3.2 krótkiego impulsu ujemnego powodującego zapisanie bajtu w pamięci.



Rys. 1 Schemat ideowy programatora

Wykaz elementów

Kondensatory:

- C1100µF/16
- C4, C2100nF
- C3470µF/25

Rezystory:

- RP1R-PACK 10kΩ
- R1100 Ω
- R2470 Ω
- R32kΩ
- R4, R5, R63,3kΩ

Półprzewodniki:

- D1LED
- D2dioda Zenera 12V
- D31N4148
- D4dioda Zenera 5,6V
- IC1, IC2PCF8574 AP
- IC37805
- IC474LS05
- T1, T2, T3BC548

Pozostałe:

- CON1 ...złącze CENTRONICS 36 lutowane w płytce
- CONARK2 (3,5mm)
- CON3Podstawka DIL20 precyzyjna

Komplet podzespołów z płytka jest dostępny w sieci handlowej AVT jako kit szkolny AVT-2502

7. Aby zweryfikować zapisane dane (bajt), należy teraz obniżyć napięcie na wejściu RST do poziomu logicznej jedynki, ustawić odpowiednią kombinację logiczną (odczyt danych) na wejściach portu P3 i dokonać odczytu danych z wyjść portu P1. (Uwaga, weryfikacja danych może być także dokonana „hurtowo”: po zapisaniu całej zawartości pamięci odczytujemy cały program i porównujemy z oryginałem znajdującym się na dysku komputera).
8. Po sprawdzeniu poprawności zapisu bajtu zwiększamy wartość wewnętrznego licznika pamięci programu o „1” przez podanie pojedynczego impulsu na wejście XTAL1.
9. Powtarzamy operacje opisane w punktach 5 do 8 aż do zapisania całej zawartości pamięci, czyli dla układu bagatelka: 2048 razy!
10. Opcjonalnie wpisujemy do pamięci procesora bity zabezpieczające. Ustawiamy stan niski na wejściu XTAL.
11. Ustawiamy stan niski na wejściu RST.
12. Odłączamy wszystkie wyprowadzenia procesora.

Funkcja	RST	P3.2	P3.3	P3.4	P3.5	P3.7
Zapis danych	12V	IMP*)	L	H	H	H
Odczyt danych	H	H	L	L	H	H
Protekcja – bit 1	12V	IMP	H	H	H	H
Protekcja – bit 2	12V	IMP	H	H	L	L
Kasowanie	12V	IMP	H	L	L	L
Odczyt sygnatury	H	H	L	L	L	L

*) IMP- ujemny impuls 1 ms (kasowanie 10 ms)

owane są z komputera poprzez cztery linie złącza CENTRONICS i przekazywane na wejścia procesora za pośrednictwem interfejsów linia I²C – wyjście równoległe, wykorzystujących popularne układy PCF8574.

Wyjaśnienia może wymagać jeszcze sprawa układu z tranzystorami T1 ... T3. Ten fragment układu nie jest niczym innym jak programowanym stabilizatorem, dostarczającym napięć potrzebnych do sterowania wejściem RST programowanego procesora. Napięcia przełączane są stanem na linii 5/12 wyprowadzonej z wyjścia 12 IC2. Stan

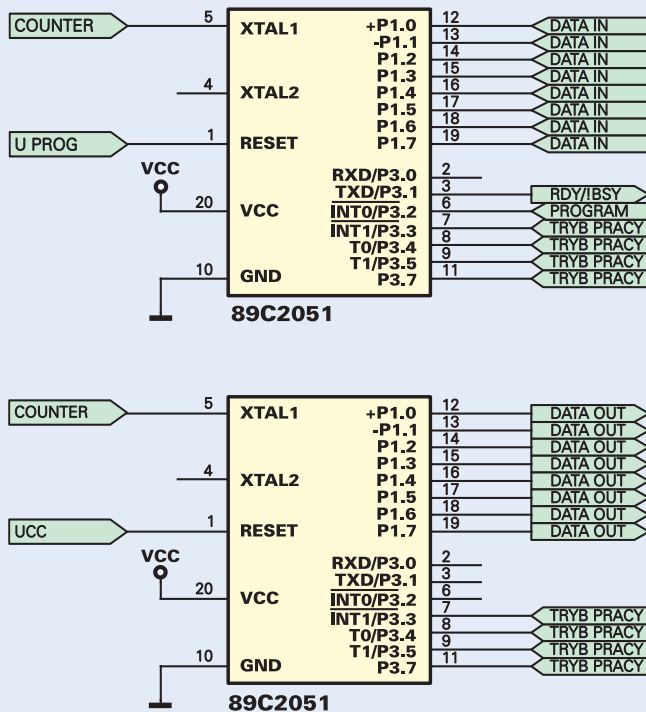
wysoki na tym wyjściu powoduje dostarczenie do wejścia RST procesora napięcia równego 5V, a stan niski pozwala na programowanie procesora poprzez wymuszenie na wejściu RST napięcia +12V. Natomiast wystąpienie na wyjściu 11 IC2 stanu wysokiego powoduje odłączenie jakiegokolwiek napięcia od wejścia RST i wyzerowanie procesora.

Dioda LED służy jedynie do wizualnej kontroli pracy programatora, włączając się podczas każdej wykonywanej przez układ operacji.

nien być wyposażony w podstawkę typu ZIF (Zero Inserting Force) umożliwiającą łatwą wymianę programowanych układów. Jednak koszt takiej podstawki przekroczy z pewnością koszt wszystkich pozostałych elementów potrzebnych do budowy programatora! Jeżeli więc macie zamiar programować procesory jedynie sporadycznie, do nie warto kupować drogiej podstawki ZIF i zamiast niej zastosować zwykłą podstawkę, byle dobrej jakości (taki element będzie dostarczany w kicie). Jeżeli jednak zdecydujecie się na podstawkę ZIF, to należy ją zakupić osobno.

Montaż układu przeprowadzamy w typowy sposób, unikając wzorowania się na modelu pokazanym na fotografii. Dlaczego? A no, przez wrodzone mi roztargnienie zamocowałem tam odwrotnie podstawkę ZIF tak, że pomimo iż układ pracował poprawnie, wkładanie procesorów do podstawki było bardzo utrudnione (dźwignia zawadzała o stabilizator napięcia i kondensator elektrolityczny).

Warto jeszcze poruszyć sprawę zasilania zmontowanego układu. Z uwagi na stosowane w urządzeniu stabilizatory napięcia, musimy dostarczyć do niego napięcie stałe z przedziału 13 ... 16VDC. Napięcie większe może spowodować uszkodzenie lub wadliwe działanie stabilizatora IC3, a mniejsze może doprowadzić do obniżenia napięcia na wejściu RST procesora poniżej 12V.



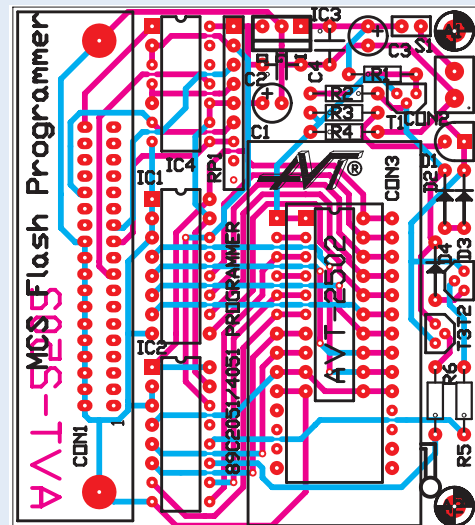
Rys. 2 Programowanie i weryfikacja

Teraz możemy już wyjąć zaprogramowany procesor z podstawki, wstawić go do uruchamianego układu, stwierdzić, że napisany podczas nieprzespanej nocy program zawiera jeszcze błędy, poprawić je i ... zacząć całą zabawę od początku! Pocięszające jest tylko to, że cały opisany wyżej algorytm programowania wykonywany jest automatycznie przez komputer w czasie nie dłuższym niż kilka sekund!

No tak, podczas opisywania algorytmu programowania procesora 89C2051 przy okazji wyczerpująco odpowiedzieliśmy na pytanie „Jak to działa?”! Wszystkie dane transmi-

Montaż i uruchomienie

Na rysunku 3 została pokazana mozaika ścieżek płytki drukowanej, wykonanej na laminacie dwustronnym z metalizacją oraz rozmieszczenie na niej elementów. Jeszcze przed rozpoczęciem montażu powinniśmy chwilę się zastanowić i powziąć ważną decyzję. Pomyśmy, jakie są nasze zamiary na przyszłość i czy chcemy opracowywać wiele układów wykorzystujących procesory 89CX051? W zasadzie każdy programator, do którego z założenia często wkładamy i wyjmujemy programowane układy powi-



Rys. 2 Schemat montażowy