

Programator procesorów AVR do kompilatora BASCOM AVR

AVT-871

Mam nadzieję, że wśród elektroników jednym z największych hitów 2000 roku będą pakiety oprogramowania BASCOM 8051 i BASCOM AVR produkcji holenderskiej firmy MCS Electronics. Posiadają one poza kompilatorem języka BASIC rozliczne programy dodatkowe, takie jak emulator sprzętowy, emulator programowy i obsługa programatorów wielu typów. Jednak jedynie z emulatora programowego możemy korzystać bez dodatkowego wyposażenia. Zarówno emulatorzy sprzętowe, jak i oczywiście obsługa programatorów wymaga zakupu lub wykonania we własnym zakresie dodatkowego hardware'u.

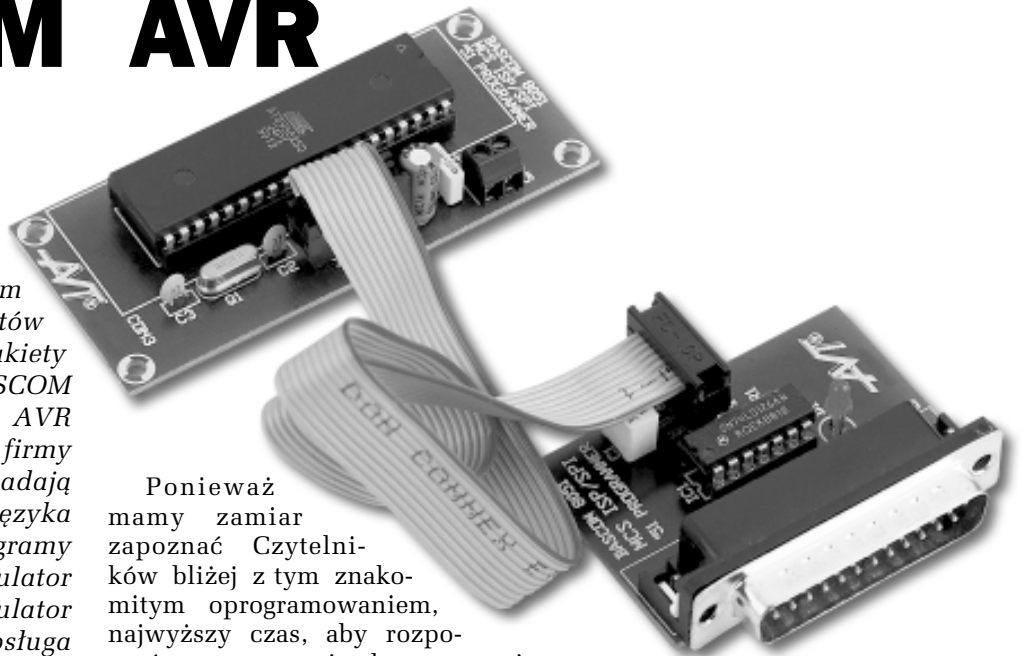
Ponieważ mamy zamiar zapoznać Czytelników bliżej z tym znakomitym oprogramowaniem, najwyższy czas, aby rozpoczynając przygotowania do stworzenia odpowiedniej bazy sprzętowej, niezbędnej do posługiwania się BASCOM-ami. W najbliższym czasie opublikowane zostaną opisy programatorów i emulatorów sprzętowych współpracujących z pakietem BASCOM 8051, a dzisiaj zajmijmy się banalnie prostym układem programatora procesorów AVR współpracującym z pakietem BASCOM AVR. Schemat tego programatora został opublikowany przez MCS Electronics, a ja pozwoliłem sobie jedynie na rozbudowanie go i wykonanie do niego płytek obwodów drukowanych.

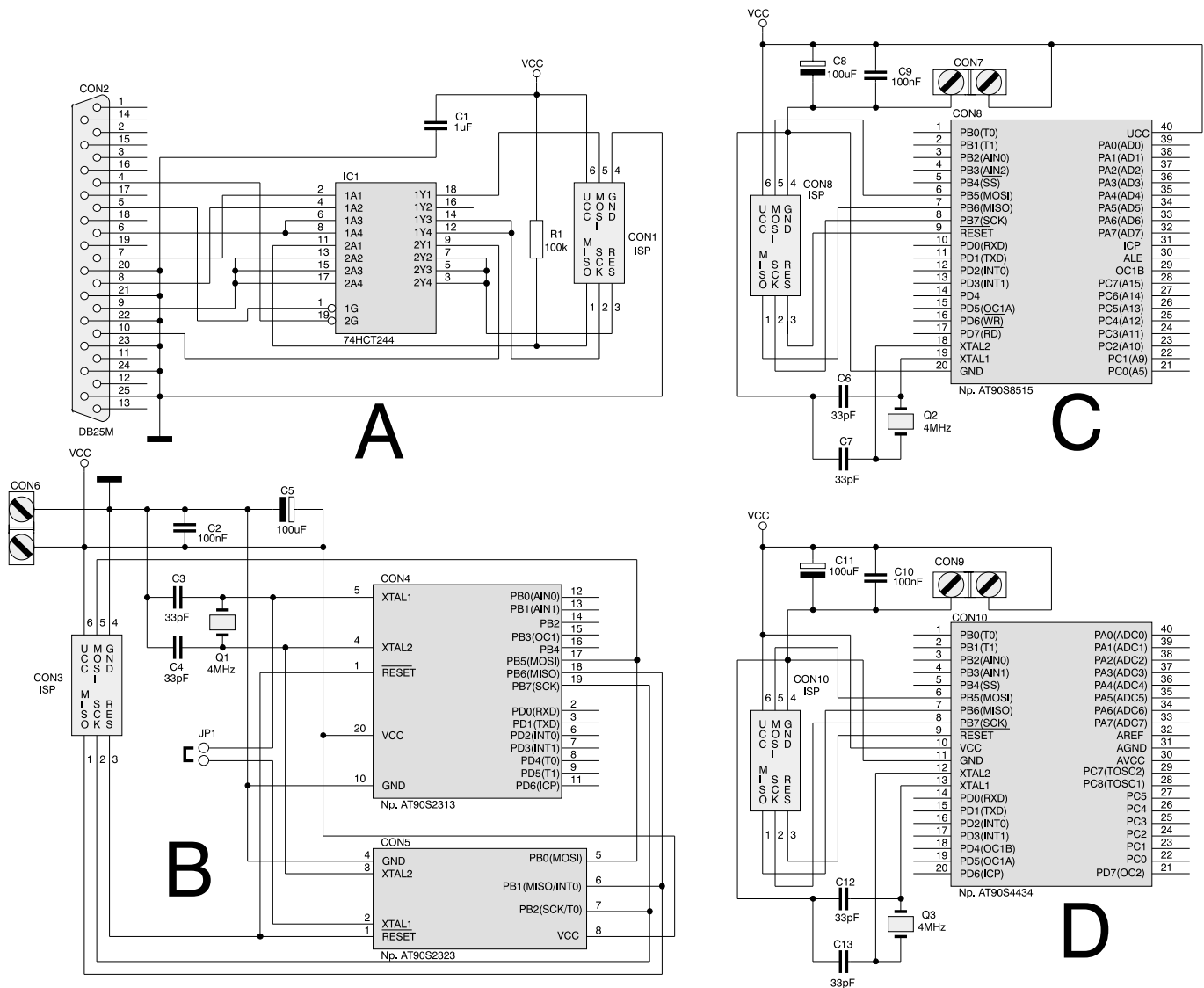
Z pewnością wielu Czytelników zapyta o sens budowy programatora związanego „na śmierć i życie” z konkretnym pakietem oprogramowania. Moim zdaniem takie działanie ma sens, i to z dwóch powodów. Po pierwsze, program BASCOM AVR jest dostępny także w darmowej wersji demo (można go ściągnąć z naszej witryny internetowej www.ep.com.pl/ftp). Wersja ta posiada wprawdzie ograniczenie długości kodu wynikowego, ale nie posiada żadnych ograniczeń kodu generowanego przez inne kompilatory i wczytywanego w celu wykorzystania w programa-

torze BASCOM-a. Tak więc, nawet pisząc programy w innym języku niż MCS-BASIC, możemy wykorzystać wszystkie możliwości proponowanego programatora posługując się darmową wersją BASCOM-a AVR tylko do programowania procesorów.

Po drugie, układ opisany w tym artykule jest zmodyfikowaną wersją programatora STK200 firmy Kanda, bardzo popularnego wśród elektroników. W Internecie można znaleźć sporo oprogramowania obsługującego ten programator, dostępnego jak shareware.

Proponowany układ może służyć programowaniu procesorów AVR w trybie programowania szeregowego wykorzystującego magistralę SPI. Procesory można programować zarówno w podstawkach wyposażonych w niezbędny rezonator kwarcowy, jak i w systemie, za pomocą specjalnego złącza ISP. Ponieważ nie wszystkie układy bazujące na procesorach AVR posiadają takie złącza, pozwoliłem sobie zaprojektować specjalne złącza umożliwiające programowanie w systemie bez konieczności dolutowywania prze-





Rys. 1. Schemat elektryczny programatora.

wodów czy dołączania ich za pomocą chwytaków.

Programator jest układem banalnie prostym i możliwym do wykonania nawet przez początkującego elektronika. Także koszt budowy urządzenia nie powinien zbyt mocno nadszarpnąć niczyjego budżetu.

Opis działania układu

Schemat elektryczny układu programatora AVR pokazano na rys. 1. Jak łatwo zauważyć, ponieważ cała inteligencja programatora skupiona została w jego części software'owej, układ został maksymalnie uproszczony i zawiera jedynie jeden aktywny element: IC1 - 74HCT244.

Na rys. 1a przedstawiono właściwy układ programatora, wyposażonego w złącze ISP transmitujące sygnały do procesora znajdującego się w jednej z podstawek

CON4, CON5, CON8 lub CON10 w drugiej części układu. Ta część, połączona z programatorem za pomocą przewodu taśmowego, zawiera także kwarc, kondensatory i zasilacz, potrzebne do zapewnienia warunków pracy procesora podczas programowania. Działanie programatora najlepiej prześledzić omawiając algorytm szeregowego programowania procesorów AVR.

Aby zaprogramować pamięć procesora, programator musi wykonać następujące czynności:

1. Podczas włączania zasilania powinien wymusić stan niski na wejściach RESET! i SCK procesora. Nie wszystkie programatory (w tym opisywany) są w stanie wykonać tę czynność i w takim przypadku konieczne jest, po wymuszeniu stanu niskiego na wejściu SCK, podanie na wejście RESET! dodat-

niego impulsu o czasie trwania dwóch cykli zegarowych.

2. Po upływie co najmniej 20 ms programator musi wysłać do procesora instrukcję zezwolenia na programowanie. Składnia tej i innych instrukcji podana jest w tab. 1.

3. Kolejną czynnością jest sprawdzenie poprawności transmisji. Po wysłaniu przez programator drugiego bajtu instrukcji Programming Enable, procesor powinien odpowiedzieć „odesłaniem“ do programatora wartości tego bajtu. Jeżeli tak się stanie, to należy uznać, że transmisja jest prawidłowa i przystąpić do wykonywania kolejnych instrukcji. Jeżeli jednak programator nie otrzymał „echa“ od procesora, to należy powtórzyć próby nawiązania transmisji. Brak „echa“ po 32 próbie świadczy o niemożności zsynchronizowania układów.

4. Po nawiązaniu transmisji programator powinien wysyłać do procesora kolejne instrukcje przewidziane dla aktualnie wykonywanego zadania. Możliwe jest zaprogramowanie zarówno pamięci danych, jak i programu, odczyt ich zawartości oraz zabezpieczenie pamięci programu za pomocą dwóch bitów zabezpieczających.

5. Po zakończeniu programowania ustawienie stanu wysokiego na wejściu RESET procesora umożliwia jego poprawną pracę.

Montaż i uruchomienie

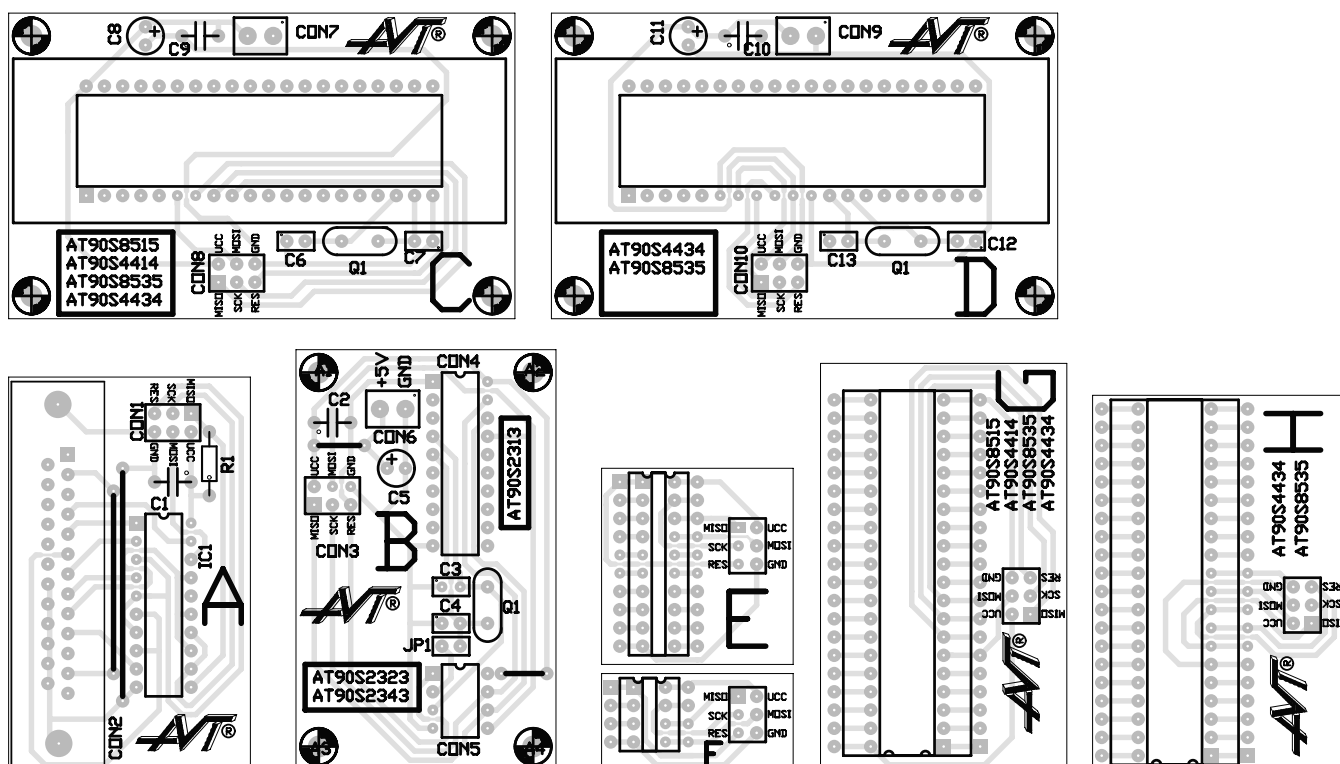
Na rys. 2 pokazano rozmieszczenie elementów na płytkach obwodów drukowanych. Wszystkie płytki zostały wykonane na laminacie jednostronnym, a ich liczba z pewnością zdziwiła wielu Czytelników. Przed rozpoczęciem montażu wyjaśnijmy sobie zatem, do czego służy każda z płytek, tym bardziej, że nie wszystkie musimy od razu montować.

Na **płytkę A** umieszczony został właściwy układ programatora wraz z wtykiem DB25 i złączem ISP o rozkładzie wyprowadzeń zgodnym z zaleceniami firmy ATMEL. **Płytkę B** zawiera dwie podstawki, kwarc z niezbędnymi kondensatorami oraz złącze ISP.

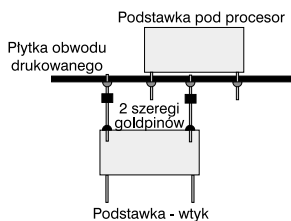
Instrukcja	Format instrukcji				Działanie
	Bajt 1	Bajt 2	Bajt 3	Bajt 4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	Zezwolenie na programowanie
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Kasowanie obydwóch pamięci
Read Program Memory	0010 H000	xxxx xaa	bbbb bbbb	oooo oooo	Odczyt górnej lub dolnej (H) części danych spod adresu a:b
Write Program Memory	0100 H000	xxxx xaa	bbbb bbbb	iiii iiii	Zapis górnej lub dolnej (H) części danych spod adresu a:b
Read EEPROM Memory	1010 0000	xxxx xxxx	xbbb bbbb	oooo oooo	Odczyt z pamięci danych spod adresu b
Write EEPROM Memory	1100 0000	xxxx xxxx	xbbb bbbb	iiii iiii	Zapis do pamięci danych pod adres b
Write Lock Bits	1010 1100	111x xABx	xxxx xxxx	xxxx xxxx	Zapis bitów zabezpieczających A i B
Read Signature Bits	0011 0000	xxxx xxxx	xxxx xxbb	oooo oooo	Odczyt typu układu o spod adresu b
Komentarz:	a - wyższe bity adresu b - niższe bity adresu H=0 - niższy bajt, H=1 - wyższy bajt o - odczyt danych i - zapis danych x - bez znaczenia A - bit zabezpieczający 1 B - bit zabezpieczający 2				

Na tej płytce możemy programować procesory AVR o 20 wyprowadzeniach (np. AT90S2313) i 8 wyprowadzeniach (np. AT90S2323 lub AT90S2343). Jumper JP1 służy do odłączania rezonatora kwarcowego w przypadku programowania procesorów AT90S2343 lub ATTINY.

Płytkę C służy do programowania procesorów o 40 nóżkach, z zasilaniem doprowadzonym do 20 i 40 wyprowadzenia, np. AT90S8515. **Płytkę D** możemy wykorzystać do prac z procesorami 40-nóżkowymi, z zasilaniem doprowadzonym do pinów 10 i 11, np. AT90S4434.



Rys. 2. Rozmieszczenie elementów na płytkach drukowanych (widok zmniejszony do 95%).

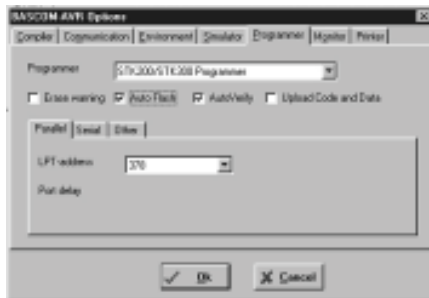


Rys. 3. Sposób montażu płytki pod procesor.

Jest oczywiste, że nie wszystkie płytki będą nam od razu potrzebne i możemy montować je dopiero w momencie, kiedy zajdzie potrzeba zaprogramowania jakiegoś konkretnego typu procesora. Montaż wszystkich płytek wykonujemy w typowy, wielokrotnie już opisywany sposób, a do połączenia ich ze sobą użyjemy przewodu taśmowego i wtyków zaciskanych typu Z-LPV10, których 4 złącza pozostaną nie wykorzystane.

Otwarta pozostaje jeszcze sprawa zastosowania pozostałych czterech płytek, oznaczonych jako E, F, G i H. Mogą one posłużyć do zbudowania prostych, ale niezwykle użytecznych gadżetów, które mogą znacznie ułatwić programowanie procesorów w gotowych układach. Programowanie ISP ma wiele zalet, lecz korzystanie z niego jest niejednokrotnie bardzo utrudnione. Rzadko kiedy możemy umieścić na płytce układu mikroprocesorowego złącze ISP i albo zmuszeni jesteśmy programować procesor poza systemem, albo dolutowywać do płytki dodatkowe przewody czy też posługiwać się miniaturowymi chwytakami teletechnicznymi. Z płytek E..H możemy w prosty sposób (rys. 3) zmontować „przejsiówki”, w których umieszczamy procesor, dołączamy do złącza ISP przewód do programatora, a całość umieszczamy w podstawce procesora uruchamianego systemu. Szczegóły tego rozwiązania są wyraźnie widoczne na fotografiach.

Jak już wiemy, opisany programator przeznaczony jest w zasadzie wyłącznie do współpracy z pakietem BASCOM AVR. Szczegółowy opis tego oprogramowania zamieścimy w kolejnych numerach Elektroniki Praktycznej, tak że podam teraz jedynie skrócony opis korzystania z naszego programatora.



Rys. 4. Okno konfiguracji programatora.

1. Po uruchomieniu programu BASCOM AVR (lub BASCOM AVR Demo) wybieramy z menu *OPTIONS* pozycję *PROGRAMMER*.

2. Ze spisu dostępnych programatorów wybieramy *STK200/STK300 programmer*

3. Ustawiamy konfigurację programatora stosownie do aktualnych potrzeb (rys. 4).

4. Jeżeli mamy zamiar programować procesor programem napisanym w MCS BASIC, to kompilujemy go (klawisz F7) i uruchamiamy programator (klawisz F4). Procesor zostanie automatycznie rozpoznany i, jeżeli uprzednio zazaczyliśmy opcję „AUTO FLASH“, zaprogramowany bez naszego udziału.

5. Jeżeli korzystamy z pliku BIN lub HEX utworzonego w innym kompilatorze, to wczytujemy ten plik do bufora (rys. 5) i programujemy nim procesor.

Zbigniew Raabe, AVT
zbigniew.raabe@ep.com.pl

WYKAZ ELEMENTÓW

Rezystory

R1: 100kΩ

Kondensatory

C1: 1μF

C2, C9, C10: 100nF

C3, C4, C6, C7, C12, C13: 33pF

C5, C8, C11: 100μF/10V

Półprzewodniki

IC1: 74HCT244

Różne

CON1, CON2: 2x5 goldpin

CON2: złącze DB25M lutowane w płytkę

CON4: podstawka precyzyjna DIL 20

CON5: podstawka precyzyjna DIL 8

JP1: 2x goldpin + jumper

Q1, Q2, Q3: rezonator kwarcowy 4MHz

2 wtyki Z-LPV10

przewód taśmowy 10-żyłowy ok. 0,5mb

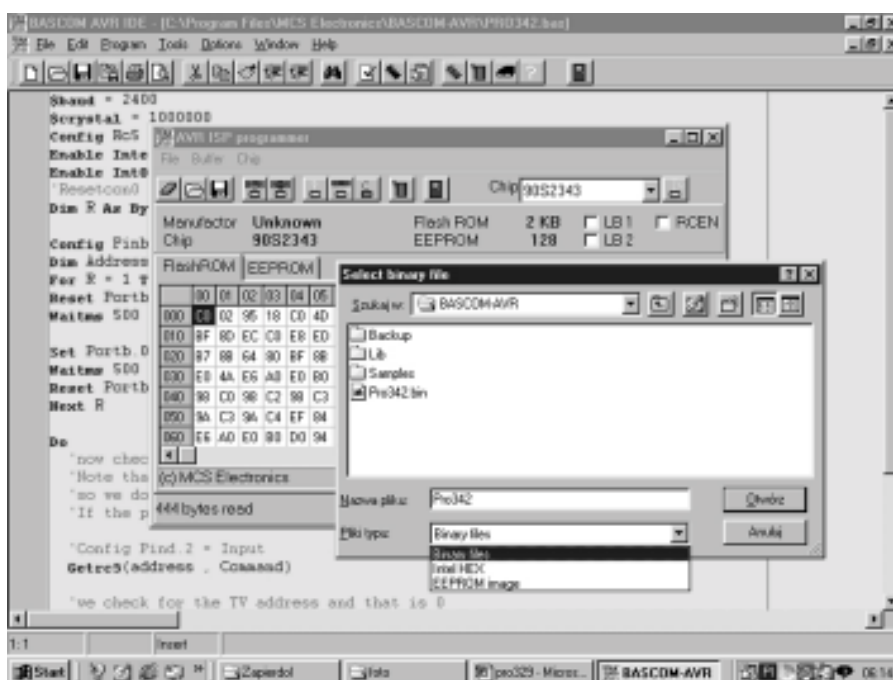
2 dodatkowe podstawki precyzyjne DIL20

3 dodatkowe podstawki precyzyjne DIL8

4 dodatkowe podstawki precyzyjne DIL40

2 listwy goldpinów 1-rzędowych prostych

Wzory płytek drukowanych w formacie PDF są dostępne w Internecie pod adresem: <http://www.ep.com.pl/pcb.html> oraz na płycie CD-EP06/2000 w katalogu PCB.



Rys. 5. Widok okien programatora ISP.