

Dariusz CABAN

Institut Informatyki Teoretycznej i Stosowanej PAN

MIKROSTEROWNIKI RODZINY MCS-51 PROGRAMOWALNE W SYSTEMIE DOCELOWYM

Streszczenie. W artykule przedstawiono wybrane mikrosterowniki rodziny MCS-51 z wewnętrzną pamięcią programu, dodatkowo wyposażone w opcję programowania szeregowego. Opcja ta umożliwia ładowanie programu do pamięci również po umieszczeniu mikrosterownika w systemie docelowym. Omówiono stosowane metody realizacji programowania szeregowego.

THE MCS-51 FAMILY MICROCONTROLLERS PROGRAMMABLE IN A TARGET SYSTEM

Summary. This article presents selected MCS-51 family microcontrollers with internal program memory, additionally equipped with serial programming option. This option enables program downloading to microcontroller already mounted in a target system. This article also presents applied methods of serial programming.

1. Wprowadzenie

Mikrosterowniki rodziny MCS-51 należą do grupy najpopularniejszych mikrosterowników 8-bitowych. Chociaż zaczęto je produkować na początku lat osiemdziesiątych, ich architektura nadal jest rozwijana. Obecnie rodzina ta obejmuje co najmniej kilkadziesiąt typów mikrosterowników, różniących się między sobą zestawem wewnętrznych zasobów. Jednym z zasobów jest pamięć programu. Zastosowanie mikrosterownika z wewnętrzną pamięcią programu przynosi wielorakie korzyści: pozwala zmniejszyć liczbę elementów w systemie docelowym, a co za tym idzie liczbę połączeń i moc pobieraną przez system, upraszcza obwody drukowane, podnosi niezawodność systemu. Jeżeli przy tym nie ma potrzeby użycia w systemie zewnętrznej pamięci danych oraz innych układów wyposażonych w taką samą magi-

stralę, liczba dostępnych linii wejścia-wyjścia powiększa się wówczas o linie portów P0 i P2. Możliwe jest ponadto zabezpieczenie zawartości pamięci programu przed nieupoważnionym odczytem.

Stosowanych jest kilka wariantów wewnętrznej pamięci programu w mikrosterownikach rodziny MCS-51. Obok mikrosterowników z pamięcią ROM o zawartości określonej maską w trakcie procesu produkcyjnego oferowane są mikrosterowniki z pamięcią programowalną przez użytkownika: EPROM (w odmianie jedno- i wielokrotnie programowalnej), FLASH oraz podtrzymywaną bateryjnie statyczną pamięcią RAM. Mikrosterowniki takie można podzielić na dwie grupy:

- programowalne tylko równoległe, tj. za pomocą programatora,
- programowalne zarówno w sposób równoległy, jak i szeregowy przy wykorzystaniu np. mikrokomputera IBM PC.

Szczególnie użyteczna jest opcja programowania szeregowego - zmiana zawartości pamięci programu nie wymaga użycia specjalizowanego urządzenia, a poza tym może być ona przeprowadzana bez potrzeby wyjmowania mikrosterownika z systemu docelowego. Przewiduje się, że w najbliższych latach mikrosterowniki posiadające taką opcję będą zajmować dominującą pozycję. Celem niniejszej publikacji jest zaprezentowanie wybranych mikrosterowników rodziny MCS-51 programowalnych w systemie docelowym oraz sposobów realizacji programowania szeregowego.

2. Mikrosterowniki programowalne przez synchroniczny interfejs SPI

Poprzez synchroniczny interfejs szeregowy SPI (ang. *Serial Peripheral Interface*) programowany jest mikrosterownik AT89S8252 firmy Atmel, będący rozbudowaną wersją układu 80C52 [2,8]. Wyprowadzeniami interfejsu są te linie portu P1, którym w układzie 80C52 nie przydzielono funkcji alternatywnych. Wbudowany sterownik interfejsu SPI można w trybie normalnej pracy mikrosterownika wykorzystywać do komunikacji z innymi mikrosterownikami i układami zewnętrznymi wyposażonymi w ten interfejs. W porównaniu z układem 80C52 mikrosterownik AT89S8252 zawiera, oprócz interfejsu SPI, następujące rozszerzenia:

- 8 kB pamięci programu typu FLASH,
- 2 kB pamięci danych typu EEPROM,
- dodatkowy rejestr wskaźnikowy DPTR,
- dodatkowe funkcje układu zegara/licznika T2,

- układ nadzorczy (ang. *watchdog*),
- możliwość blokowania generacji sygnału ALE,
- częstotliwość zegarowa w zakresie $0 \div 24$ MHz,
- napięcie zasilania w zakresie 4÷6 V.

Gwarantowana przez producenta trwałość pamięci FLASH wynosi 1000 przeprogramowań, natomiast pamięci EEPROM - 100000. Pamięci EEPROM przyporządkowano adresy z zakresu $0 \div 7FFH$ w tej samej przestrzeni adresowej, którą zajmuje zewnętrzna pamięć danych. Dostęp do obu rodzajów pamięci jest realizowany za pomocą rozkazu MOVX, a ich rozróżnienie następuje poprzez odpowiednie ustawienie jednego z bitów sterujących w obszarze rejestrów specjalnego przeznaczenia.

Kiedy układ zegara/licznika T2 pracuje jako generator taktowania dla standardowego układu transmisji szeregowej, istnieje możliwość wyprowadzenia na wyjście P1.0 fali prostokątnej o wypełnieniu 1/2 i częstotliwości uzależnionej od częstotliwości zegarowej mikrosterownika i ustawionej prędkości transmisji.

Układ nadzorczy został zrealizowany jako układ zliczający impulsy, których źródłem jest niezależny wewnętrzny generator współpracujący z programowalnym dzielnikiem częstotliwości. Przy nominalnej częstotliwości generatora wynoszącej 62.5 Hz oraz współczynniku podziału o postaci 2^n , gdzie $n = 0, 1, \dots, 7$, otrzymuje się zakres maksymalnych czasów odświeżania układu nadzorczego od 16 do 2048 ms. W rzeczywistości występują odchylenia wartości czasu odświeżania od wartości zaprogramowanej dochodzące do $\pm 30\%$.

Interfejs SPI został opracowany przez firmę Motorola dla realizacji połączeń między układami scalonymi w systemach mikrokomputerowych. Interfejs może sprzęgać wiele układów, przy czym jeden z nich ma pozycję uprzywilejowaną i nosi nazwę układu nadrzędnego (ang. *master*), pozostałe zaś są układami podrzędnymi (ang. *slave*). Informacja jest przesyłana za pośrednictwem trzech linii (podano ich oznaczenia stosowane przez producenta mikrosterownika AT89S8252): impulsów zegarowych SCK (ang. *Serial Clock*), służących do synchronizacji informacji, linii danych wejściowych dla układów podrzędnych MOSI (ang. *Master Out Slave In*) oraz linii danych wyjściowych z układów podrzędnych MISO (ang. *Master In Slave Out*). Do każdego układu podrzędnego jest ponadto doprowadzona osobna linia wyboru \overline{SS} (ang. *Slave Select*); wybór za pomocą logicznego stanu wysokiego jest rzadziej stosowany. Do zadań układu nadrzędnego należy inicjacja i sterowanie przebiegiem komunikacji z wybranym układem podrzędnym oraz wytwarzanie impulsów zegarowych, niezależnie od tego, czy w danej chwili jest nadawcą czy odbiorcą informacji [4]. Mikrosterownik AT89S8252 w trybie programowania jest traktowany jako układ podrzędny, natomiast w trybie normalnej pracy może być skonfigurowany jako układ nadrzędny lub podrzędny. Dla każ-

dego z wymienionych przypadków inne są funkcje wyprowadzonych na zewnątrz mikrosterownika linii interfejsu SPI, co pokazano w tabeli 1.

Tabela 1

Funkcje linii interfejsu SPI mikrosterownika AT89S8252 w różnych trybach pracy

linia interfejsu SPI mikrosterownika	tryb programowania mikrosterownika	tryb normalnej pracy mikrosterownika	
		jako układ nadrzędny	jako układ podrzędny
\overline{SS} (P1.4)	nieużywana	nieużywana	wyбір układu
MOSI (P1.5)	wejście	wyjście	wejście
MISO (P1.6)	wyjście	wejście	wyjście
SCK (P1.7)	wejście	wyjście	wejście

Częstotliwość zegarowa mikrosterownika AT89S8252 podczas programowania szeregowego powinna mieścić się w zakresie 4+24 MHz, a częstotliwość impulsów zegarowych przesyłanych linią SCK nie może przekroczyć 1/40 częstotliwości zegarowej. Do programowania służy siedem 3-bajtowych rozkazów, których format przedstawiono w tabeli 2. Bajty 1 i 2 są przesyłane linią MOSI i reprezentują kod rozkazu, natomiast bajt 3 zawiera daną, przy operacji odczytu przesyłaną linią MISO. Mikrosterownik zapamiętuje stan linii danych wejściowych w chwilach wyznaczanych przez narastające zbocza impulsów zegarowych, natomiast kolejne bity odczytywanej z niego informacji są wystawiane na linię danych wyjściowych przy zboczach opadających. Przejście w tryb programowania jest wymuszony poprzez ustawienie na wejściu zerowania RST mikrosterownika logicznego stanu wysokiego i wysłanie rozkazu rozpoczęcia programowania. Zapisywany bajt zarówno pamięci FLASH, jak i

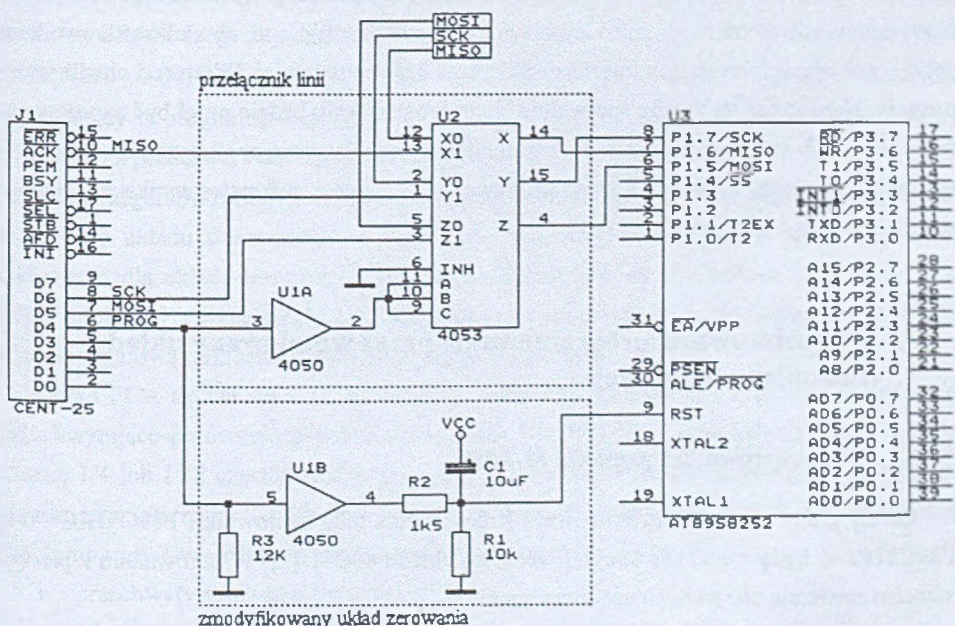
Tabela 2

Zestawienie rozkazów programowania szeregowego mikrosterownika AT89S8252 (a_i - bity adresowe, d_i - bity danych, lb_i - bity zabezpieczające, programowane, jeśli $lb_i = 0$, x - wartość bitu nieistotna)

Rozkaz	Format rozkazu		
	bajt 1	bajt 2	bajt 3
rozpoczęcie programowania	10101100	01010011	xxxxxxxx
kasowanie całej zawartości obu pamięci	10101100	xxxxx100	xxxxxxxx
odczyt bajtu pamięci programu	$a_{12}a_{11}a_{10}a_9a_8001$	$a_7a_6a_5a_4a_3a_2a_1a_0$	$d_7d_6d_5d_4d_3d_2d_1d_0$
zapis bajtu pamięci programu	$a_{12}a_{11}a_{10}a_9a_8010$	$a_7a_6a_5a_4a_3a_2a_1a_0$	$d_7d_6d_5d_4d_3d_2d_1d_0$
odczyt bajtu pamięci danych	$00a_{10}a_9a_8101$	$a_7a_6a_5a_4a_3a_2a_1a_0$	$d_7d_6d_5d_4d_3d_2d_1d_0$
zapis bajtu pamięci danych	$00a_{10}a_9a_8110$	$a_7a_6a_5a_4a_3a_2a_1a_0$	$d_7d_6d_5d_4d_3d_2d_1d_0$
programowanie bitów zabezpieczających	10101100	$lb_1lb_2lb_3xx111$	xxxxxxxx

EEPROM, jest najpierw automatycznie kasowany, dzięki czemu można zmieniać również zawartość tylko wybranego fragmentu pamięci. Maksymalny czas zapisu bajtu wynosi 2.5 ms przy napięciu zasilania mikrosterownika równym 5V. Istnieje możliwość uzyskania informacji o stanie zaawansowania tej operacji - w tym celu należy dokonać odczytu bajtu i porównać odczytaną wartość z zapisywaną. Pozytywny wynik porównania wskazuje na to, że zapis został zakończony. Możliwe jest także kasowanie całej zawartości obu pamięci. Zajmuje ono ok. 16 ms, próba odczytu w tym czasie dowolnego bajtu da w wyniku wartość 00H. Zakończenie programowania mikrosterownika AT89S8252 następuje poprzez ponowne ustawienie stanu niskiego na jego wejściu zerowania, po czym mikrosterownik rozpocznie wykonywanie programu od adresu 0000H [2].

Do realizacji szeregowego programowania mikrosterownika AT89S8252 wystarcza użyć np. portu drukarki mikrokomputera IBM PC i odpowiedniego programu ładującego (ang. *loader*). Użytkownik może na podstawie informacji zawartych w [1,2] zaimplementować taki program samodzielnie. Może też skorzystać z oprogramowania udostępnianego bezpłatnie w sieci Internet, np. *AT89S8252 In System Programmer V1.00*, opracowanego przez firmę AEC Electronics (<http://www.aec-electronics.co.nz/>).



Rys. 1. Przelącznik linii interfejsu SPI mikrosterownika pomiędzy układami zewnętrznymi a portem drukarki mikrokomputera IBM PC oraz zmodyfikowany układ zerowania mikrosterownika

Fig. 1. The switching circuit of microcontroller's SPI lines between peripherals and IBM PC printer port and modified reset circuit of microcontroller

Jeżeli planowane jest wykorzystanie linii interfejsu SPI mikrosterownika w systemie docelowym, wówczas konieczne staje się ich przełączanie pomiędzy układami zewnętrznymi a mikrokomputerem programującym. Ponieważ funkcje tych linii zmieniają się w zależności od trybu, w jakim aktualnie pracuje mikrosterownik (tabela 1), przełącznik powinien umożliwiać dwukierunkowy przepływ sygnałów. Jego przykładową realizację przedstawiono na rysunku 1. Wykorzystano w niej niedrogi i łatwo dostępny układ CMOS typu 4053, zawierający trzy niezależnie adresowane 1-bitowe multipleksery/demultipleksery analogowe [6]. Widoczne na rysunku przyporządkowanie linii portu drukarki mikrokomputera IBM PC poszczególnym liniom interfejsu SPI jest identyczne z przyjętym przez twórców wymienionego wyżej programu. Zalecają oni także, aby sygnał PROG, wymuszający zmianę trybu pracy mikrosterownika, był podawany na jego wejście zerowania RST poprzez opornik $1.5k\Omega$. Przeprowadzone próby wykazały jednak, że taka modyfikacja układu zerowania mikrosterownika jest niewystarczająca. Przy $PROG = 1$ napięcie na wejściu RST nie osiągało wymaganej wartości potrzebnej do zmiany trybu. Poprawę uzyskano po zastosowaniu dodatkowej bramki buforowej 4050 [6]. Wejście tej bramki przyłączono poprzez opornik do masy, przez co po zakończeniu programowania stan niski na wejściu zerowania mikrosterownika zostanie utrzymany także po odłączeniu systemu docelowego od mikrokomputera programującego. Projektant oprogramowania użytkowego musi też w tym przypadku zadbać o to, aby mikrosterownik nie dokonywał operacji na liniach interfejsu SPI przez czas co najmniej 500 ms od chwili wyzerowania. Naruszenie tej zasady spowoduje, że mikrosterownik będzie mógł być zaprogramowany w sposób szeregowy tylko jednokrotnie. Ponowne włączenie opcji programowania szeregowego wymaga wówczas kasowania zawartości pamięci mikrosterownika za pomocą specjalizowanego programatora [3].

3. Mikrosterowniki programowalne przez wbudowany układ transmisji szeregowej

3.1. Mikrosterowniki z pamięcią FLASH

Do tej grupy należą oferowane przez firmę Philips mikrosterowniki: P89C51RC+ oraz P89C51RD+, będące rozbudowanymi wersjami układu 80C51 [7]. W porównaniu z pierwowzorem zawierają one następujące rozszerzenia:

- pamięć programu typu FLASH o pojemności 32 kB w P89C51RC+ oraz 64 kB w P89C51RD+,
- pamięć ROM o pojemności 1 kB,

- dodatkowa pamięć danych RAM o pojemności 384 bajtów w P89C51RC+ oraz 896 bajtów w P89C51RD+,
- dodatkowy rejestr wskaźnikowy DPTR,
- układ zegara/licznika T2,
- układ PCA (ang. *Programmable Counter Array*),
- dodatkowe funkcje układu transmisji szeregowej,
- układ nadzorczy,
- możliwość blokowania generacji sygnału ALE,
- częstotliwość zegarowa w zakresie 0÷33 MHz.

Minimalna trwałość pamięci FLASH wynosi 1000 przeprogramowań. Pamięć ROM nakłada się na pamięć programu począwszy od adresu FC00H; w razie potrzeby można ją zablokować, udostępniając programowi użytkowemu pełną przestrzeń adresową. Zawiera ona program ładujący oraz funkcje realizujące operacje na pamięci FLASH, które mogą być wywoływane także przez program użytkowy. Dodatkowa wewnętrzna pamięć RAM jest zorganizowana w dwa bloki. Pierwszy, o pojemności 128 bajtów, umieszczono w tym samym obszarze adresowym co blok rejestrów specjalnego przeznaczenia (adresy 80H÷7FH), jest on dostępny wyłącznie w trybie adresowania pośredniego. Drugi blok stanowi tzw. pamięć rozszerzoną ERAM (ang. *Expanded RAM*), która współdzieli obszar o adresach: 00H÷FFH w P89C51RC+ i 00H÷2FFH w P89C51RD+ z zewnętrzną pamięcią danych. Rozróżnienie, czy występujący w programie rozkaz MOVX dotyczy pamięci ERAM czy pamięci zewnętrznej, następuje na podstawie wartości odpowiedniego bitu sterującego.

Układ zegara/licznika T2 w omawianych mikrosterownikach to rozszerzona wersja analogicznego układu stosowanego w mikrosterowniku 8052. Przy pracy w trybie generatora taktowania dla układu transmisji szeregowej możliwe jest wyprowadzenie na wyjście P1.0 fali prostokątnej o wypełnieniu 1/2 i częstotliwości określonej przez dwa parametry: częstotliwość zegarową mikrosterownika oraz prędkość transmisji.

Układ PCA składa się z 16-bitowego licznika współpracującego z pięcioma modułami przechwytywająco-porównującymi. Licznik może być taktowany impulsami o częstotliwości równej 1/4 lub 1/12 częstotliwości zegarowej mikrosterownika, częstotliwości przepelnienia układu zegara/licznika T0 lub impulsami podawanymi z zewnątrz. Każdy z modułów jest programowany niezależnie do pracy w jednym z czterech trybów:

- przechwytywania zdarzeń zewnętrznych,
- odmierzania czasu,
- generowania impulsów,
- modulacji wypełnienia impulsów.

Jeden z modułów przechwytyjąco-porównujących może być ponadto wykorzystywany jako układ nadzorczy o ustawianym przez użytkownika maksymalnym czasie odświeżania.

Dodatkowe funkcje układu transmisji szeregowej obejmują wykrywanie błędnych bitów stopu (tzw. błąd ramki, ang. *frame error*) oraz automatyczne rozpoznawanie adresów w trybie pracy wieloprocesorowej.

Jak wyżej wspomniano, w charakterze układu nadzorczego może być używany jeden z modułów układu PCA. Oprócz tego przedstawiane mikrosterowniki są wyposażone w specjalizowany układ nadzorczy. Układ ten ma postać 14-bitowego licznika zliczającego cykle maszynowe. Maksymalny czas odświeżania jest w tym przypadku stały i równy $16383 \cdot \text{czas trwania cyklu}$.

Przejsięcie w tryb programowania szeregowego następuje po wyzerowaniu mikrosterownika przy $\overline{\text{PSEN}} = 0$, $\text{ALE}/\overline{\text{PROG}} = 1$ (linia ta może też być nie podłączona) oraz napięciu na wejściu $\overline{\text{EA}} / \text{VPP}$ wyższym od napięcia zasilania mikrosterownika. Na początku mikrokomputer programujący powinien wysłać znak bajt z kodem ASCII litery 'U', co umożliwi mikrosterownikowi samoczynne ustalenie prędkości transmisji w łączy szeregowym. Dalej mikrosterownik przyjmuje już tylko informacje w postaci rekordów INTEL HEX. Stosowanych jest sześć typów rekordów, ich zawartość i znaczenie są opisane w [7]. Jeżeli wystąpiły błędy przy transmisji rekordu, mikrosterownik odsyła bajt 'X'. Poprawny odbiór rekordu w większości przypadków jest potwierdzany bajtem '.'. Wyjątek stanowi rekord zawierający bajty do zapisu w pamięci FLASH. Mikrosterownik odsyła kod znaku kropki po poprawnym odebraniu rekordu i pomyślnym zakończeniu operacji zapisu pamięci; błędy zapisu są sygnalizowane bajtem 'R'. Wyjście z trybu programowania powinno zostać poprzedzone zapisem wartości 0 do tzw. bajtu statusowego, dzięki czemu mikrosterownik po sprzętowym zerowaniu rozpocznie wykonywanie programu od adresu 0000H.

Program ładujący o nazwie WinISP, przeznaczony dla mikrokomputerów IBM PC, jest dostępny bezpłatnie na stronie internetowej firmy Philips (<http://www.semiconductors.philips.com>).

3.2. Mikrosterowniki z podtrzymywaną bateryjnie pamięcią SRAM

Reprezentantem tej grupy mikrosterowników jest DS5000T firmy Dallas Semiconductor [5]. Jest to rozbudowana wersja układu 8051 i w porównaniu z nim zawiera następujące rozszerzenia:

- 8 lub 32 kB nieulotnej pamięci SRAM, w której mogą być przechowywane zarówno program, jak i dane,
- układ zegara czasu rzeczywistego,

- układ nadzorczy,
- układ monitorowania napięcia zasilającego,
- częstotliwość zegarowa do 16 MHz.

Nieulotność zawartości pamięci SRAM zapewnia bateria litowa, a gwarantowany czas podtrzymania wynosi ok. 10 lat. W trakcie programowania mikrosterownika użytkownik ustala konfigurację tej pamięci za pomocą dwóch parametrów: RANGE oraz PARTITION. Wartość pierwszego z parametrów jest równa pojemności pamięci nieulotnej, natomiast drugi określa, jaka jej część, począwszy od adresu 0000H, ma być traktowana jako pamięć programu (0, 2, ..., 8 kB przy RANGE = 8 kB, 0, 2, ..., 28, 32 kB przy RANGE = 32 kB). Pozostała część jest widziana jako zewnętrzna pamięć danych.

Układ zegara czasu rzeczywistego umożliwia określenie bieżącego czasu oraz daty. Układ ten działa niezależnie od jednostki centralnej mikrosterownika i jest taktowany impulsami o częstotliwości 32768 Hz, wytwarzanymi przez dedykowany generator. Informacja o czasie i dacie, reprezentowana w formacie BCD, jest przechowywana w bloku ośmiu rejestrów 8-bitowych. Komunikacja z układem zegara jest realizowana za pomocą rozkazów MOVX, przy czym musi być ustawiony odpowiedni bit sterujący. Zapis/odczyt zawartości rejestrów nie odbywa się jednak w sposób równoległy, lecz szeregowy. Bity informacji zapisywanej są przesyłane poprzez linię adresową A0, bity informacji odczytywanej poprzez linię danych D7, natomiast rodzaj operacji określa stan linii adresowej A2; na pozostałych liniach adresowych powinien być stan niski. Niemożliwe są operacje na pojedynczym rejestrze, zapisywanie lub odczytywanie musi być cały blok. Ponadto musi to być poprzedzone zapisem 64-bitowego ciągu o ustalonej postaci. Tak więc odwołanie do bloku rejestrów układu zegara czasu rzeczywistego wymaga zrealizowania 128 rozkazów MOVX. Po zaniku napięcia zasilającego układ kontynuuje odmierzenie czasu dzięki wbudowanej w mikrosterownik baterii litowej.

Układ nadzorczy tworzy licznik zliczający cykle maszynowe. Wewnętrzny sygnał zerowania mikrosterownika jest generowany po 122880 cyklach.

Zadaniem układu monitorowania napięcia zasilającego V_{CC} jest wykrywanie zmian wartości tego napięcia i podejmowanie odpowiednich czynności przy przekroczeniu wartości progowych. Wyróżnione są trzy wartości progowe: V_{PFW} , V_{CCMIN} , V_{LI} , między którymi zachodzi relacja $V_{PFW} > V_{CCMIN} > V_{LI}$. Spadek napięcia zasilającego poniżej V_{PFW} jest sygnalizowany jednostce centralnej mikrosterownika za pomocą żądania przerwania; przerwanie to ma najwyższy priorytet. Przy minimalnej wartości V_{CC} następuje przejście w stan wstrzymania pracy mikrosterownika poprzez wygenerowanie wewnętrznego sygnału zerowania i wyłączenie oscylatora. Mikrosterownik pozostaje w tym stanie do chwili, gdy ponownie $V_{CC} > V_{CCMIN}$. Przy $V_{CC} < V_{LI}$ z baterii litowej zasilana jest pamięć wewnętrzna oraz układ zegara czasu rzeczywistego.

W celu uaktywnienia trybu szeregowego programowania mikrosterownika DS5000T należy najpierw ustawić na linii RST stan wysoki, a na linii $\overline{\text{PSEN}}$ stan niski; zalecane jest także, aby linie P2.6 oraz P2.7 przerwano lub został na nich wymuszony stan wysoki. Następnie mikrosterownik musi odebrać od mikrokomputera programującego znak CR kodu ASCII. Umożliwi to mikrosterownikowi samoczynne ustalenie prędkości transmisji w łączu szeregowym. Nowa zawartość pamięci wewnętrznej jest przesyłana w postaci rekordów w formacie INTEL HEX. Prowadzona jest przy tym na bieżąco weryfikacja zapisywanych bajtów. Stosowany format znaków to 8-N-1 (8 bitów informacyjnych, brak kontroli parzystości, 1 bit stopu). Wyjście z trybu programowania następuje poprzez odtworzenie połączeń linii P2.6, P2.7 i $\overline{\text{PSEN}}$ oraz ustawienie stanu niskiego na wejściu zerowania, po czym mikrosterownik rozpoczyna realizację programu począwszy od adresu 0000H.

Firma Dallas Semiconductor udostępnia bezpłatnie na swojej stronie internetowej (<http://www.dalsemi.com>) program ładujący o nazwie KIT, przeznaczony dla mikrokomputerów IBM PC.

4. Podsumowanie

Do realizacji wewnętrznej pamięci programu w mikrosterownikach coraz częściej stosowana jest pamięć kasowalna i programowalna elektrycznie. Niektóre mikrosterowniki zawierające taką pamięć mogą być programowane na dwa sposoby: równoległe oraz szeregowo. Pierwszy sposób wymaga użycia specjalizowanego programatora, w drugim wystarczy tylko mikrokomputer. Opcja programowania szeregowego daje jeszcze jedną korzyść - zmiana zawartości pamięci wewnętrznej może zostać przeprowadzona nawet wtedy, gdy mikrosterownik znajduje się już w systemie docelowym. Upraszcza to proces uruchamiania systemu, jak i wymianę oprogramowania w przypadku jego rozbudowy. W niniejszym artykule przedstawiono kilka wybranych mikrosterowników rodziny MCS-51, wyposażonych w opcję programowania szeregowego. Ich cena nie jest wygórowana, np. AT89S8252 można kupić już za kilkanaście złotych, a odpowiednie programy ładujące dostępne są bezpłatnie w sieci Internet. Stanowią więc one interesujące uzupełnienie popularnej rodziny mikrosterowników, a fakt, że do ich programowania nie są potrzebne specjalizowane urządzenia, może mieć dla niektórych osób istotne znaczenie.

LITERATURA

1. Anderson P. H.: Use of a PC Printer Port for Control and Data Acquisition. Artykuł dostępny w sieci Internet na stronie <http://et.nmsu.edu/~etti/fall96/computer/printer/printer.html>.
2. Atmel Corp.: AT89S8252. 8-bit Microcontroller with 8K Bytes Flash. Materiały informacyjne dostępne w sieci Internet na stronie <http://www.atmel.com/>.
3. Atmel Corp: AT89S8252, AT89LS8252, AT89S53, AT89LS53 Errata Sheet. Materiały informacyjne dostępne w sieci Internet na stronie <http://www.atmel.com/>.
4. Caban D.: Synchroniczne interfejsy szeregowo dla sprzęgania układów w systemach mikrokomputerowych. Artykuł zgłoszony do publikacji w Archiwum Informatyki Teoretycznej i Stosowanej.
5. Dallas Semiconductor: Secure Microcontroller User's Guide. Materiały informacyjne dostępne w sieci Internet na stronie <http://www.dalsemi.com/>.
6. Motorola: CMOS Logic Data. DL131 Rev. 1., 1988.
7. Philips: P89C51RC+/P89C51RD+. 80C51 8-bit Flash microcontroller family. 32K/64K ISP FLASH with 512-1K RAM. Materiały informacyjne dostępne w sieci Internet na stronie <http://www.semiconductors.philips.com/>.
8. Starecki T.: Mikrokontrolery jednocukładowe rodziny 51. NOZOMI, Warszawa 1996.

Recenzent: Dr inż. Wojciech Mielczarek

Wpłynęło do Redakcji 26 czerwca 2000 r.

Abstract

Application of microcontroller with internal program memory simplifies structure of a target system. Several variants of such memory are used: ROM, EPROM, FLASH and battery-backed SRAM. These microcontrollers can be divided into two groups:

- programmable only parallelly,
- programmable both parallelly and serially.

Especially serial programming option is convenient to an user, because it enables program downloading to microcontroller already mounted in the target system. Beside, popular IBM PC microcomputer is sufficient for downloading, loaders are available without any fee in Internet. This article presents selected MCS-51 family microcontrollers programmable serially.

The first described microcontroller is AT89S8252, manufactured by ATMEL. This device is equipped with SPI interface, which can work in two modes: normal and programming. In normal mode SPI interface enables communication with external serial devices: EEPROMs, RTCs, A/D and D/A converters, LED and LCD controllers. In programming mode it enables downloading. Internal memory of AT89S8252 is programmed by seven instructions listed in table 2. When using of SPI interface in the target system is planned, switching of interface lines between IBM PC and serial devices is necessary; appropriate switching circuit is shown in figure 1.

Next, microcontrollers programmable through standard asynchronous serial port were presented: P89C51RX+, manufactured by Philips and DS5000T, offered by Dallas. Former is equipped with FLASH memory, latter with battery-backed SRAM. Both microcontrollers are programmed using INTEL HEX records.