

Internetowy interfejs dla mikrokontrolera, część 1

AVT-5055



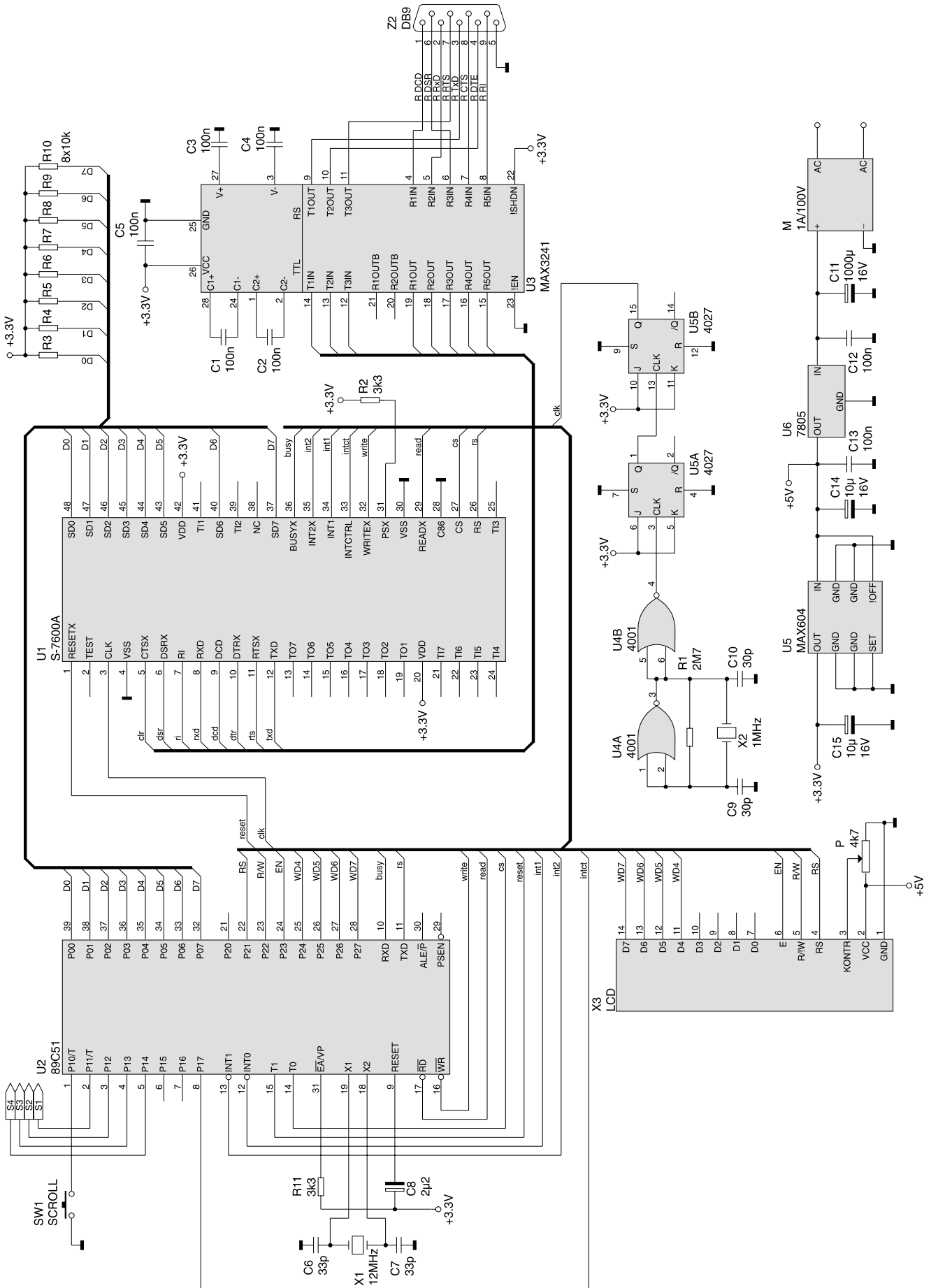
Jesteś ciekaw jak za pomocą 8-bitowego mikrokontrolera poszaleć w Internecie? Chciałbyś samodzielnie zbudować serwer internetowy, odbierać i nadawać e-maile? Chcesz nadążyć za modnymi zakamarkami współczesnej techniki mikroprocesorowej?

W artykule opisujemy sposób wykorzystania ultranowoczesnego układu scalonego, który spełnia rolę sprzętowego stosu TCP/IP.

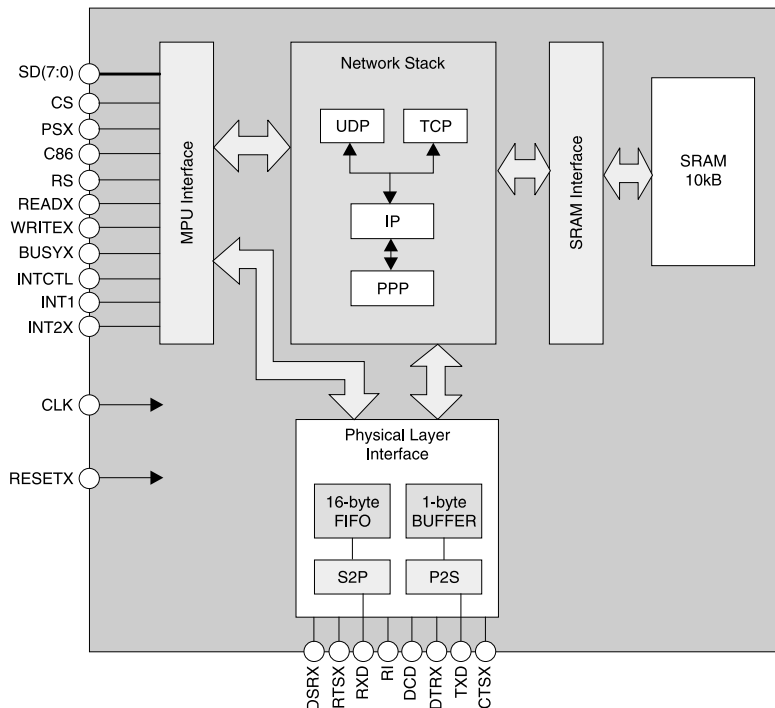
O możliwości wykorzystania Internetu do przesyłania danych przez mikroprocesorowe sterowniki słyhać coraz częściej i coraz głośniej. Firmy produkujące mikrokontrolery oferują zestawy ewaluacyjne pozwalające na połączenie się z Siecią - większość z nich opisywaliśmy już w EP. Na przykład takimi spektakularnymi aplikacjami mogą być opisywane już opracowania Atmela, Ziloga, Microchipsa czy firmy Ubicom (dawniej Scenix). W tym ostatnim wykorzystano nowo opracowane mikrokontrolery z odpowiednio wydajnym rdzeniem oraz „dużą” pamięcią danych i programu. Przy takiej koncepcji, obsługa wszystkich protokołów sieciowych niezbędnych do połączenia z Internetem i przesyłania informacji realizowana jest programowo.

Takie rozwiązanie ma oczywiście wiele zalet. Po poznaniu modułów programowych systemu, można w miarę elastycznie dostosowywać go do własnych wymagań. Jednak z drugiej strony, implementacja sieciowa zajmuje dużo mocy obliczeniowej i pamięci mikrokontrolera. Nie bez znaczenia jest też fakt, że trzeba stosować nowe mikrokontrolery, a to wiąże się zawsze z wprowadzeniem nowych narzędzi (kompilatorów, programatorów, emulatorów itp.), a więc ze zwiększeniem kosztów.

Cóż zatem zrobić, aby nie narazić się na dodatkowe koszty i nie popaść w uzależnienie od konkretnego mikrokontrolera? Od tych dylematów uwalnia nas oferta firmy Seiko - układ S7600A. Jest to specjalizowany układ



Rys. 1. Schemat elektryczny interfejsu.



Rys. 2. Schemat blokowy układu S7600A.

umożliwiający szybkie i bezproblemowe połączenie z siecią, praktycznie dowolnego, obecnie produkowanego mikrokontrolera. Przykładem niech będzie opisywany tutaj system, w którym doskonale wszystkim znany i niezbyt wydajny mikrokontroler AT89C51 wraz z układem S7600A pracuje jako klient poczty elektronicznej.

Opis układu

Schemat interfejsu pokazano na rys. 1. Najważniejszym jego elementem jest oczywiście S7600A - układ wielkiej skali integracji (VLSI) zawierający w swojej strukturze kompletny, sprzętowy stos TCP/IP wraz z zaimplementowanym protokołem PPP, interfejs łączy szeregowego UART z 16-

bitowym odbiorczym buforem FIFO, 10kB pamięci RAM oraz rozbudowany szeregowo - równoległy interfejs MPU do połączenia z mikrokontrolerem (rys. 2). Przez ten interfejs mikrokontroler zapisuje lub odczytuje informacje do/z wewnętrznych rejestrów S7600A.

Układ zasilany jest napięciem o wartości z zakresu 2,4V...3,6V i pobiera minimalny prąd w czasie pracy: 0,9mA w trakcie transmisji i tylko 150µA w stanie oczekiwania na transmisję. Tak niski pobór mocy wskazuje na to, że konstruktorzy przewidywali jego pracę przy zasilaniu bateryjnym. Układ jest taktowany zewnętrznym sygnałem zegarowym. Producent zaleca częstotliwość 256KHz, ale maksymalna jej wartość może wynosić nawet 5MHz.

Układ S-7600A zawiera dwa interfejsy MPU: równoległy i szeregowy. W trybie interfejsu równoległego można połączyć magistrale danych rodziny x80 firmy Intel lub 68K firmy Motorola. Poprzez te interfejsy następuje wymiana informacji pomiędzy mikrokontrolerem a układem. Jak widać zadano, aby maksymalnie ułatwić pracę projektantom. Wybór interfejsu równoległego następuje poprzez wymuszenie wysokiego poziomu na wejściu PSX. Poziom wysoki na wejściu C86 określa tryb pracy interfejsu dla

Tab. 1. Sposób obsługi rejestrów układu S7600A dla dwóch możliwych konfiguracji interfejsu.

RS	Motorola R/WX	Intel		Funkcja
		READX	WRITEX	
1	1	0	1	Czytanie rejestru
1	0	1	0	Zapis rejestru
0	1	0	1	Czytanie rejestru indeksowego
0	0	1	0	Zapis rejestru indeksowego

procesorów Motoroli, a poziom niski dla procesorów Intela. W tab. 1 zawarto zestawienie stanów logicznych na wejściach sterujących zapisem i odczytem dla obu rodzajów magistrali.

Ponieważ w projekcie wykorzystywana jest magistrala Intela, to ją postaram się opisać dokładniej. Zainteresowani sterowaniem S7600A poprzez magistralę Motoroli mogą znaleźć odpowiednie dane w dokumentacji firmowej.

Tryb magistrali Intela jest wprowadzany, kiedy na wejściu C86 jest poziom niski „L“, a na PSX poziom wysoki „H“. Dane oraz magistrala adresowa są multipleksowane. Każdy cykl rozpoczyna się od ustawienia na magistrali adresu. Adres ten jest zatraskiwany w wewnętrznej rejestrze podczas narastającego zbocza WRITEX. Poziom niski na RS wskazuje, że strobowanie WRITEX dotyczy fazy adresu na magistrali. W następnej fazie dane mogą być zapisywane lub odczytywane poprzez wygenerowanie odpowiednich zboczy sygnałów WRITEX lub

List. 1. Procedura zapisu danych do układu S7600A przez magistralę równoległą.

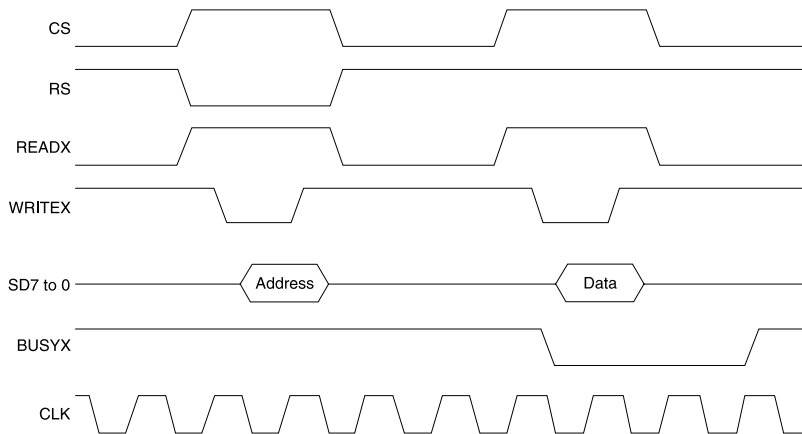
```

void zapis_ichip (unsigned char adres, unsigned char dana)
{
    cs=1;
    rs=0; //adres na magistrali
    readx=1;
    writex=0;
    P0=adres; //adres na magistrale
    writex=1;
    cs=0;
    rs=1; //dane na magistrali
    readx=0;
    cs=1;
    readx=1;
    writex=0;
    P0=dana; //dana na magistrale
    writex=1;
    cs=0;
    readx=0;
    while(busyxx==0); //czekaj na nieaktywne busyx
}
    
```

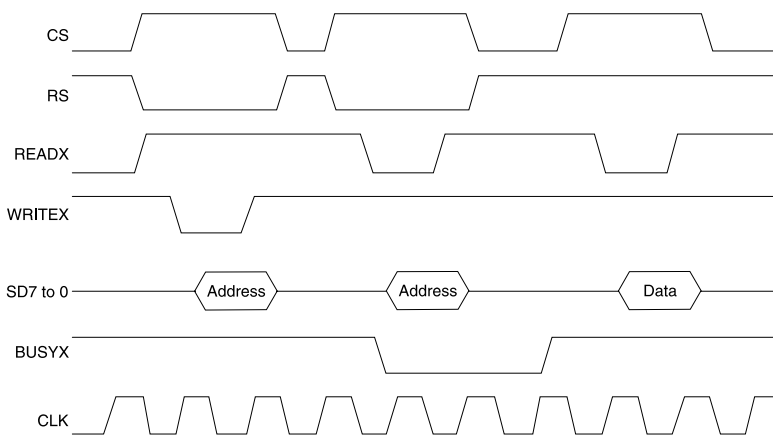
List. 2. Procedura odczytu danych z układu S7600A przez magistralę równoległą.

```

unsigned char odczyt_ichip (unsigned char adres)
{
    unsigned char dana;
    unsigned char dana_p;
    cs=1;
    rs=0;
    readx=1;
    writex=0;
    P0=adres; //adres wpisany do
    writex=1;
    cs=0;
    rs=1;
    cs=1;
    rs=0;
    P0=0xff; //ustaw jako wejsciowy
    readx=0;
    dana_p=P0; //odczytanie adresu
    readx=1;
    rs=1;
    cs=0;
    while(busyxx==0); //czekaj na nieaktywne busyx
    cs=1;
    readx=0;
    dana=P0; //odczytanie danych
    readx=1;
    while(busyxx==0); //czekaj na nieaktywne busyx
    cs=0;
    return (dana);
}
    
```



Rys. 3. Przebiegi czasowe sygnałów podczas zapisu do interfejsu równoległego.



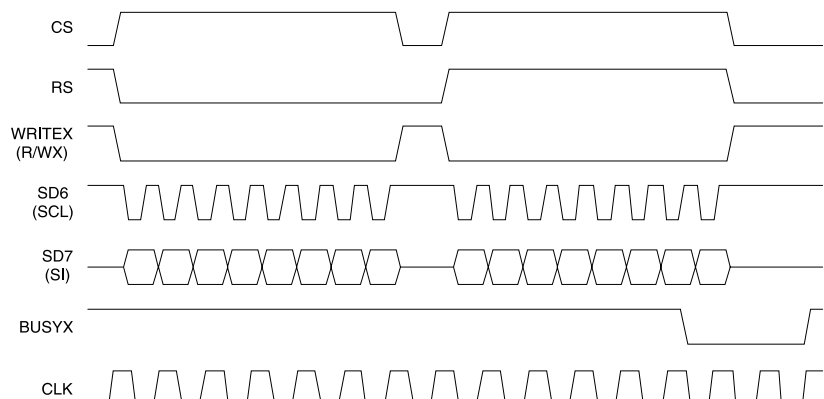
Rys. 4. Przebiegi czasowe sygnałów podczas odczytu do interfejsu równoległego.

READX. Układy logiczne interfejsu generują w tej fazie sygnał BUSYX po opadającym zboczku sygnału WRITEX lub READX. Sygnał ten staje się nieaktywny, gdy S7600A zakończy operację zapisu lub odczytu. Mikrokontroler powinien próbować sygnał BUSYX. Może on zainicjować kolejny cykl zapisu/odczytu dopiero wtedy, gdy BUSYX staje się nieaktywny.

Przebiegi czasowe dla zapisu i odczytu przez magistrale równoległą pokazano na rys. 3 i 4. W prezentowanym projekcie obsługa magistrali jest realizowana programowo. Procedury zapisu i odczytu w języku C przedstawione są na list. 1 i list. 2.

Jak już wspominałem, oprócz interfejsu równoległego można wykorzystać też interfejs szeregowy. Pozwala to połączyć S7600A z mikrokontrolerami za pomocą niewielkiej liczby linii, co ma ogromne znaczenie na przykład dla mikrokontrolerów podobnych do PIC16F84. Ten tryb jest wy-

bierany przez wymuszenie na wejściu PSX poziomu niskiego. Linia SD6 magistrali danych jest wtedy wejściem sygnału zegarowego. Linia SD5 to wejście danych, natomiast SD7 wyjście danych (patrząc od strony S7600A). Kierunkiem przepływu danych steruje wejście WRITEX. Poziom wysoki na wejściu WRITEX oznacza odczyt danych, a poziom niski zapis danych. Przebiegi czasowe inter-



Rys. 5. Przebiegi czasowe sygnałów podczas odczytu dla interfejsu szeregowego.

Tab. 2. Przestrzeń adresowa Banku 0.

Adres	Rozmiar	Zawartość
0x0000-0x07ff	2k	Kieszeń 0 bufora odbioru
0x0800-0x0bff	1k	Kieszeń 0 bufora nadawczego
0xc000-0x0fff	1k	Dane bazowe TCP
0x1000-0x13ff	1k	Bufor IP

Tab. 3. Przestrzeń adresowa Banku 1.

Adres	Rozmiar	Zawartość
0x0000-0x07ff	2k	Kieszeń 1 bufora odbioru
0x0800-0x0bff	1k	Kieszeń 1 bufora nadawczego
0xc000-0x0fff	1k	Bufor PPP
0x1000-0x13ff	1k	Bufor PAP

fejsu szeregowego pokazano na rys. 5 (cykl zapisu) i 6 (cykl odczytu).

W strukturze układu S7600A umieszczony jest kompletny port szeregowy UART. Tor odbiorczy zawiera 16-bajtowy bufor FIFO. Dane przesyłane asynchronicznie mają następujący format: 1 bit startu, 8 bitów informacyjnych i 1 bit stopu, bez bitu parzystości.

Zasadniczym blokiem układu jest jednak sprzętowy stos TCP/IP. Zawiera on moduły TCP/UDP, moduł IP, oraz moduł PPP. Z protokołami TCP/UDP/IP/ i PPP są związane 2 kieszenie umieszczone w wewnętrznej pamięci RAM. Cóż to takiego te kieszenie? Otóż są to obszary pamięci RAM, w których umieszcza się dane do przesyłania za pomocą protokołu TCP/IP. Oprócz kieszeni, w pamięci RAM podzielonej na banki po 5kB umieszczone są bufony pomocnicze protokołów TCP, IP, oraz PPP. Podział pamięci pokazano w tab. 2 i 3. Może się zdarzyć, że przy takim podziale pamięci

Tab. 4. Rejestry układu S7600A.

Adres	Rejestr	Definicja bitu								
0x00	Revision	Numer wersji rdzenia S-7600A								
0x01	General Control	-	-	-	-	-	-	-	SW_RST	
0x02	General Socket Location	0	0	0	0	0	0	S1	S0	
0x04	Master Interrupt	-	-	-	-	-	PT_INT	Link_INT	Sock_IN	
0x08	Serial Port Config	S_DAV	DCD	Dsr Hwfc	CTS	RI	DTR	RTS	SCTL	
0x09	Serial Port Int	PT_Int	-	-	-	-	-	-	-	
0x0a	Serial Port Int Mask	PINT_EN	DSINT_EN	-	-	-	-	-	-	
0x0b	Serial Port Data	Rejestr danych portu szeregowego								
0x0c 0x0d	Baud Rate Div	Rejestry określające prędkość transmisji								
0x10- 0x13	Our IP Address	Adres IP serwera dostępowego								
0x1c	Clock Div Low	Rejestr Clock Divider								
0x1d	Clock Div High									
0x20	Index	Rejestr indeksowy kieszeni								
0x21	TOS	Pole TOS								
0x22	Socked Config status Low	T0	Buff Emty	Buff Full	D_A /RST	-	-	Protocol Type		
0x23	Socked Status Mid	URG	RST	Term	Conu	Stan TCP				
0x24	Socked Activate	-	-	-	-	-	-	S1	S0	
0x26	Socked Interrupt	-	-	-	-	-	-	I1	I0	
0x28	Socked Data Avail	-	-	-	-	-	-	Dav1	Dav0	
0x2a	Socked Interrupt Mask Low	T0 En	Buff_E En	Buff_Full	DataA En	-	-	-	-	
0x2b	Socked Interrupt Mask High	Urg_En	RST-En	Term_En	ConU_En	-	-	-	-	
0x2c	Socked Interrupt Low	T0	Buff_Empty	Buff_Full	Data_Avail	-	-	-	-	
0x2d	Socked Interrupt High	URG	RST	Term	Conu	-	-	-	-	
0x2e	Socked Data	8-bitowe dane kieszeni								
0x30	TCP data Send	Wpisanie dowolnej wartości rozpoczyna wysyłanie danych								
0x30- 0x31	Buffer Out Lenght	Wielkość bufora wyjściowego (dla czytania tych rejestrów)								
0x32- 0x33	Bufer In	Wielkość bufora wejściowego (dla czytania tych rejestrów)								
0x34- 0x35	Urgent Data Pointer	Wskaźnik ważnych danych w buforze wejściowym/ wielkość datagramu UDP								
0x36- 0x37	Their port	Numer portu docelowego								
0x38- 0x39	Our port	Numer portu źródłowego								
0x3a	Socket Status High	-	-	-	-	-	-	-	Snd- bsy	
0x3c- 0x3f	Their IP address	Adres IP docelowy								
0x60	PPP Control Status	PPP_Int	Con_Val	Use_PAP	T0_Dis	PPP_Int En	Kick	PPP_En	PPP_Up SRset	
0x61	PPP Interrupt Code	Kod błędu PPP								
0x62	PPP Max Retry	-					Maks. liczba powtórzeń config request			
0x64	PPP String	Nazwa użytkownika i hasło								

mogą występować konflikty przy dostępie, tzn. 2 lub więcej modułów może chcieć w tym samym momencie odwoływać się do jednego wspólnego obszaru. Zadaniem arbitra dostępu do pamięci (rys. 7) jest właśnie bezkonfliktowe przydzielanie jej zasobów do poszczególnych modułów stosu.

Układ S-7600A zawiera dwa 5-kilobajtowe banki pamięci (0 i 1), jak to pokazano w tab. 2 i 3.

Rejestry wewnętrzne S-7600A są podzielone na 3 grupy: globalne, bezpośrednie i indeksowe. Rejestry globalne zajmują przestrzeń adresową od adresu 0x00 do 0x1d, oraz od 0x60 do 0x6f. Pośrednie

WYKAZ ELEMENTÓW

Rezystory

R1: 2,7MΩ
R2...R11: 3,3kΩ
Potencjometr 4,7kΩ

Kondensatory

C1...C5, C12, C13: 100nF
C6, C7: 33pF
C8: 2,2μF/16V
C9, C10: 33pF
C11: 1000μF/16V
C14, C15: 10μF/16V

Półprzewodniki

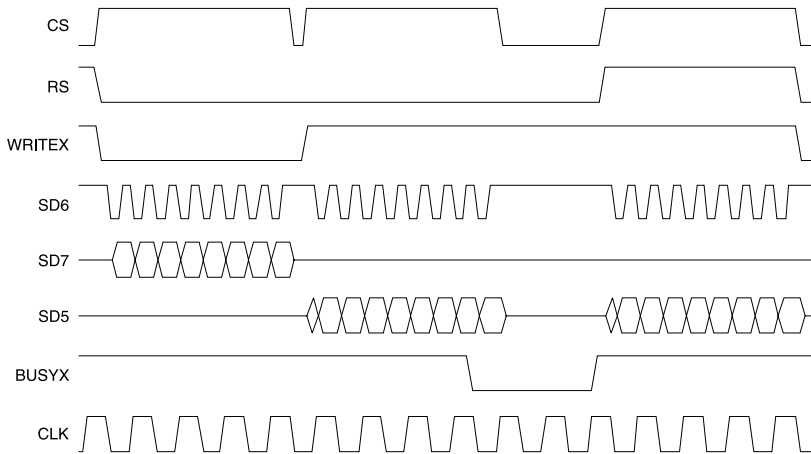
M1: 1A/100V
U1: S7600A
U2: 89C51 - zaprogramowany
U3: MAX3241
U4: CD4001
U5: CD4027
U6: MAX604
U7: 7805

Różne

X1: rezonator kwarcowy 12MHz
X2: rezonator kwarcowy 1MHz
Wyświetlacz alfanumeryczny 2x20 znaków
Z1 Złącze szufladowe 9-pinowe
Podstawka 40 DIL
SW1 przycisk typu switch
Płytką drukowana

i bezpośrednie rejestry zajmują przestrzeń od adresu 0x20 do 0x3f. Użycie rejestrów indeksowych wymaga wcześniejszego zdefiniowania indeksu kieszeni. Zależnie od tej definicji, dane w rejestrach indeksowych dotyczą kieszeni 1 lub 2. W tab. 4 pokazano zestawienie wszystkich rejestrów S7600A.

Układ S7600A jest przystosowany do fizycznego połączenia z Internetem za pośrednictwem modemu. Interfejs warstwy fizycznej wyposażony jest w związku z tym we wszystkie sygnały sterujące łącza RS232, potrzebne do prawidłowej współpracy z modemem. Sygnały te mają poziomy napięcie standardu TTL, a jak wiadomo modemy wymagają poziomów zgodnych ze standardem RS232. Odpowiedni konwerter zbudowany jest w oparciu o układ U3 MAX3241. Zasilanie układu S7600A napięciem +3,3V wymusiło zastosowanie konwertera również zasilanego tym napięciem. Złącze Z2 jest 9-pinowym męskim złączem szufladowym. Sygnały na Z2 dołączone są do jego pinów



Rys. 6. Przebiegi czasowe sygnałów podczas zapisu dla interfejsu szeregowego.

zgodnie ze standardem stosowanym w komputerach PC. Można bez problemu podłączyć do układu dowolny modem zewnętrzny za pomocą standardowego kabla używanego do połączenia modemu z komputerem.

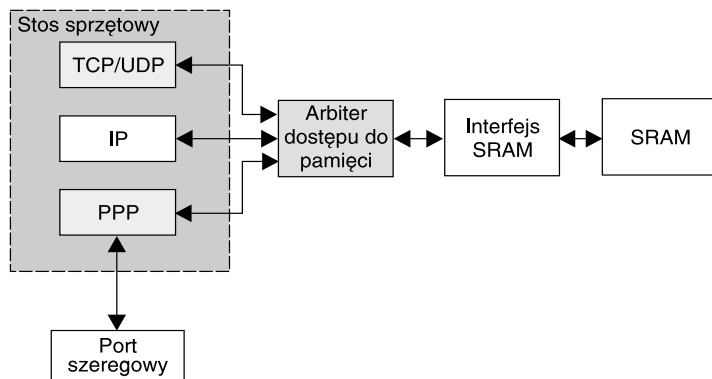
Sygnal o częstotliwości 1MHz, wytwarzany w generatorze zbudowanym ze zlinearyzowanej bramki U4A i rezonatora X2, jest następnie dzielony przez 4 w dwu przerzutnikach U5A i U5B. Układy U4 i U5 są również zasilane napięciem +3,3V. Prostokątny przebieg o częstotliwości 250kHz i amplitudzie zbliżonej do napięcia zasilania podawany jest na wyrowadzenie 3 U1.

Mikrokontroler U2 też jest zasilany napięciem +3,3V. Do portu P2 dołączony jest wyświetlacz alfanumeryczny 2x20 znaków. Dość trudno jest znaleźć taki wyświetlacz zasilany obniżonym napięciem, dlatego zastosowano popularny wyświetlacz zasilany napięciem +5V. Linie portów mikrokontrolera zasilanego napięciem niższym niż +5V mogą być „podciągane” do +5V bez szkody

dla układu. Stabilizator U6 7805 dostarcza napięcia +5V, a układ U5 MAX604 napięcia +3,3V.

Montaż układu

Płytką drukowana interfejsu pokazana jest na rys. 8. Układ S7600A jest umieszczony w 48-pinowej obudowie typu QFP przystosowanej do montażu powierzchniowego. Niestety przylutowanie układu jest dość trudne. Odległość między nóżkami obudowy wynosi tylko 0,5mm! Przed lutowaniem układ należy przykleić do płytki drukowanej, najlepiej klejem typu Poxipol i dokładnie ustawić nóżki układu na polach lutowniczych, nie zapominając o prawidłowej ich kolejności. Po związaniu kleju można przystąpić do lutowania. Trzeba się wyposażyć w lutownicę z odpowiednio cienkim grotem i dobrą lupę, najlepiej na statywie. Montaż nie jest łatwy, ponieważ obudowy QFP



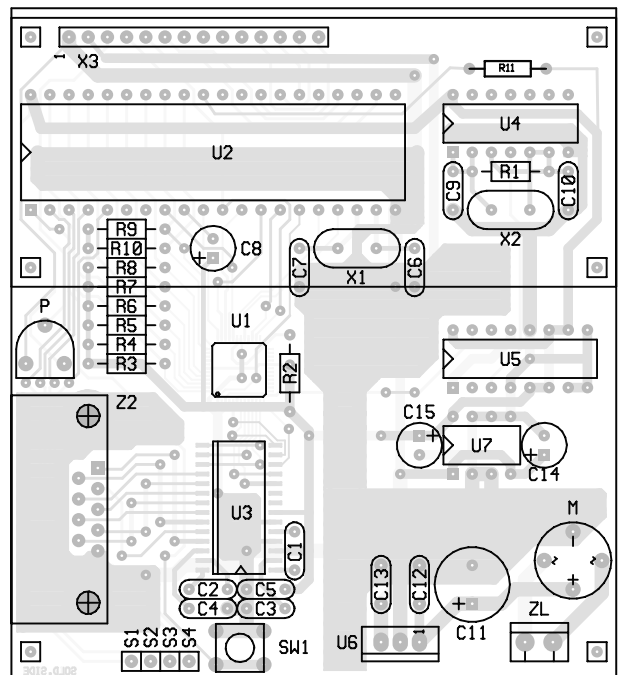
Rys. 7. Ilustracja kontrolowanego dostępu do pamięci.

zaprojektowano do montażu automatycznego, gdzie jest możliwe zachowanie wysokiej precyzji pozycjonowania i jakości lutowania.

Po zakończeniu lutowania trzeba dokładnie sprawdzić, czy nóżki S7600A są dobrze przylutowane. Z doświadczenia wiem, że niektóre luty trzeba poprawiać kilka razy. Poprawki trzeba robić bardzo delikatnie, bo łatwo uszkodzić cienkie ścieżki lub wygiąć bardzo delikatne nóżki układu. Układ U3 jest produkowany również tylko w obudowie przystosowanej do montażu powierzchniowego, ale jego montaż nie nastęcza takich problemów jak to jest w przypadku S7600A. Rozstaw nóżek i ich grubość są zdecydowanie większe. Układ U3 również najlepiej jest wstępnie przykleić do płytki, zwracając uwagę na ustawienie nóżek na polach lutowniczych. Płytką jest tak zaprojektowana, że U3 trzeba przylutować na umownej stronie lutowania (pod spodem płytki). Montaż pozostałych elementów nie powinien sprawiać kłopotów.

Tomasz Jabłoński, AVT
tomasz.jablonski@ep.com.pl

Wzory płytek drukowanych w formacie PDF są dostępne w Internecie pod adresem: <http://www.ep.com.pl/?pdf/marzec02.htm> oraz na płycie CD-EP03/2002B w katalogu PCB.



Rys. 8. Rozmieszczenie elementów na płytce drukowanej.