ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1992

Seria: INFORMATYKA z. 20 Nr kol. 1190

Stanisław WIDEŁ

KONSTRUKCJA STEROWNIKA SIECI ETHERNET OPARTEGO NA KOPROCESORZE SIECI LOKALNEJ INTEL 82586

Streszczenie. W artykule omówiono zasady konstrukcji oraz możliwe rozwiązania sterowników sieci lokalnych opartych na koprocesorze Intel 82586. Ponadto zaproponowano klasyfikację konstrukcji sterowników sieci lokalnych.

DESIGNING LOCAL AREA NETWORK CONTROLLER BASED ON 82586 COPROCESSOR

Summary. In this paper I consider designing local area network controller based on 82586 controller. As a result I propose the classification of local area network controllers.

CONSTRUCTION DES CONTRÔLEURS POUR RÉSEAUX LOCAL AVEC L'USAGE DU COPROCESSEUR 82586

Résumé. Dans ce travail, il a été presenté une procedure de la construction du contrôleurs pour réseaux local avec l'usage le coprocesseur Intel 82586. Le résultate est la classification des contrôleurs pour réseaux local.

1. Wprowadzenie

Postęp technologii wytwarzania układów scalonych z punktu widzenia architektury mikrokomputera powoduje wzrost mocy obliczeniowej jednostki centralnej, co przyjęto (głównie w reklamach) za podstawowy parametr mikrokomputera.

Jednocześnie oprócz przyśpieszenia przetwarzania, w procesorze wzrasta stopień autonomiczności i złożoności funkcji realizowanych przez układy peryferyjne. W ten sposób, co jest również bardzo ważne dla zwiększenia mocy obliczeniowej mikrokomputera, odciąża się jednostkę centralną od bezpośredniego wykonywania pewnych zadań. Pozwala to na równoległe w stosunku do operacji wejścia-wyjścia przetwarzanie informacji.

Aktualnie w typowym mikrokomputerze wykorzystuje się układy scalone o złożoności porównywalnej z mikroprocesorem. Układy te pracują samodzielnie wykonując swe specjalizowane operacje w pełni autonomicznie, zgodnie z odpowiednią listą rozkazów zadawaną zwykle przez procesor. Istnieją również układy zwane procesorami lub koprocesorami określonej funkcji. Dla przykładu w katalogu firmy Intel już w roku 1985 znajdziemy następujące koprocesory: koprocesor do obliczeń matematycznych stanowiący uzupełnienie mikroprocesora o funkcje sprzętowej realizacji obliczeń arytmetycznych, koprocesor do przetwarzania tekstu oraz koprocesor sieci lokalnej.

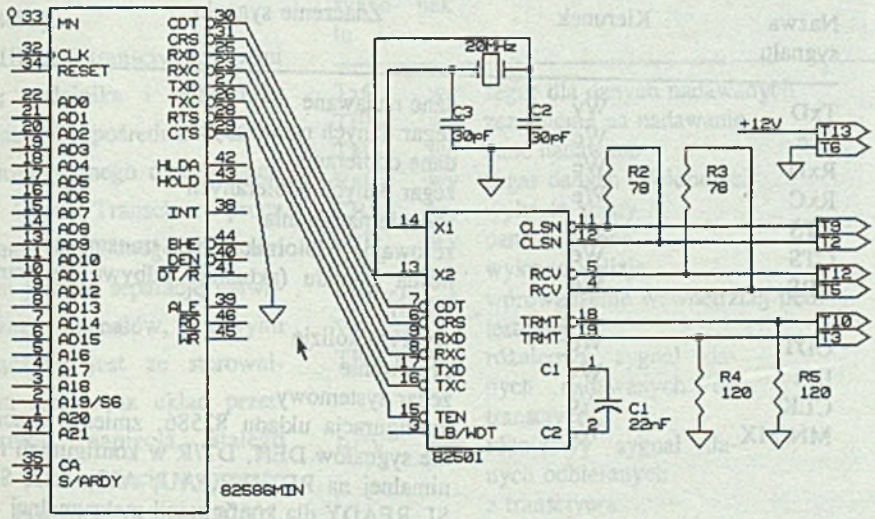
Tendencja do zwiększania autonomiczności i złożoności układów peryferyjnych wydaje się być miarą postępu elektroniki na równi ze zwiększaniem szybkości przetwarzania w kolejnych generacjach mikroprocesorów. Opinię tę potwierdzają aktualnie wprowadzane nowe podzespoły sterujące pracą układów peryferyjnych, np.: rastrowych monitorów graficznych lub układów komunikacyjnych.

Do dnia dzisiejszego podział funkcji pomiędzy specjalizowane moduły jest jedyną metodą równoległego przetwarzania informacji stosowaną w mikrokomputerach. Pomimo prowadzenia na świecie intensywych badań nad maszynami wieloprocesorowymi, sukces komercyjny, przynajmniej do tej pory, w zakresie komputerów osobistych odnoszą konstrukcje oparte na typowej architekturze von Neumana. Jeżeli ograniczymy się do mikrokomputerów, to patrząc z ekonomicznego punktu widzenia, koszt wprowadzenia wielu mikroprocesorów w jakimś modelu nie jest znaczący. Jednak, jak dotychczas, jest to niemożliwe bez ściśle określonego podziału realizowanych zadań pomiędzy poszczególne mikroprocesory.

Funkcje oraz kierunki rozwoju dla typowego systemu są dobrze określone (np: iteracyjna praca z użytkownikiem, pełne wykorzystanie możliwości graficznych, zwiększenie mocy obliczeniowej, komunikacja z zasobami lokalnymi i zdalnymi), i dlatego w coraz szerszym zakresie funkcje te realizowane są przez specjalizowane układy scalone. Niewątpliwie do takich funkcji należy zaliczyć dostęp do zdalnych zasobów poprzez sieć.

Dla wielu użytkowników sieci są jedynym rozwiązaniem pozwalającym na zorganizowanie przetwarzania rozproszonego i podziału zasobów przy oparciu się na aktualnie dostępnych handlowo mikrokomputerach z klasyczną von neumanowską architekturą.

Należy spodziewać się, że pojawienie się bazy elementowej realizującej warstwę fizyczną oraz warstwę dostępu do łącza wg modelu OSI/ISO dla sieci Ethernet zapewne spowoduje, odpowiednio do ciągłego spadku cen podzespołów, włączenie sterownika sieci do stałego wyposażenia mikrokomputerów. Należy również spodziewać się, że tendencją



Rys. 1. Typowy schemat aplikacyjny dla koprocesora sieci lokalnej Intel 82586 i szeregowego nadajnika-odbiornika sieci Ethernet Intel 82501

Fig. 1. Typical application design for local area network Intel 82586 and Ethernet serial interface Intel 82501

rozwojową dla tego typu urządzeń będzie autonomiczna realizacja coraz wyższych warstw sieciowych (stosownie do prac standaryzacyjnych w tej dziedzinie).

2. Baza elementowa firmy Intel do realizacji sterownika sieci lokalnej

Firma Intel realizując sieć lokalną typu Ethernet zaprojektowała i wprowadziła na rynek trzy elementy scalone: koprocesor sieci lokalnej oznaczony symbolem 82586, szeregowy sprzęg z siecią o oznaczeniu 82501 oraz układ nadajnika odbiornika sygnału do oraz z medium transmisyjnego, nazywany również transciverem o oznaczeniu 82502. Wypro-

Tabela 1

Opis wyprowadzeń koprocesora sieci lokalnej Intel 82586

Nazwa sygnału	Kierunek	Znaczenie sygnału
TxD	Wy	dane nadawane
Txc	We	zegar danych nadawanych
RxD	We	dane odbierane
RxC	We	zegar danych odbieranych
RTS	Wy	żądanie nadawania
CTS	We	zerowanie odbiornika przed transmisją
CRS	We	nośna sygnału (aktualnie odbywa się transmisja)
CDT	We	wykryto kolizję
INT	Wy	przerwanie
CLK	We	zegar systemowy
MN/MX	We	konfiguracja układu 82586, zmienia znaczenie sygnałów DEN, DT/R w konfiguracji minimalnej na RD, WR, AU, A22, A23, SO, SI, READY dla konfiguracji maksymalnej
AD0-AD16	We/Wy	Multipleksowane adresy i dane
A16-A18	Wy	adresy
HOLD	Wy	żądanie dostępu do pamięci
HLDA	We	potwierdzenie przydzielenia pamięci
CA	We	inicjacja wykonania rozkazów przez 82586
BHE	Wy	oznacza bardziej znaczący bajt danych
ALE	Wy	na magistrali AD0-AD16 podany jest adres
DEN	Wy	na magistralę podane są dane
SRDY/ARDY	We	wprowadza dodatkowe takty oczekiwania
RD	Wy	cykl odczytu
WR	Wy	cykl zapisu
DT/R	Wy	oznacza kierunek transmisji w wypadku użycia innych nadajników niż 8286/8287

wadzenia układów i typowy schemat aplikacyjny 82586 i 82501 przedstawiono na rysunku 1 i w tabelach 1, 2.

Inne firmy wprowadziły również układy scalone spełniające funkcje podobne do omawianych elementów firmy Intel, jak AMD (7990, 7992, 7995, 7996), MOSTEK (MK

68590, MK 68591), NATIONAL SEMICONDUCTOR (DP 8390, DP 8391, DP 8392), SEEQ (8023, 8003) WESTERN DIGITAL (WD 83C690, WD83C691, WD 83B692).

Układ transcivera pełni rolę nadajnika i odbiornika sygnału bezpośrednio z kabla koncentrycznego o rezystancji 50 Ohm. Transciver prócz elementu scalonego zawierać musi jeszcze separację galwaniczną sygnałów, którymi połączony jest ze sterownikiem sieci oraz układ przetwornicy napięcia stałego +12 V na napięcia wymagane do zasilania układu 82502. Norma sieci Ethernet pozwala na zainstalowanie transcivera jako osobnego urządzenia. Sterownik z transciverem połączony jest czterema parami skrętek dla sygnałów Nadawania, Odbierania, Kolizji i napięcia 12 V zasilającego układ transcivera. Długość kabla nie może przekroczyć 50 metrów.

Tabela 2

Opis wyprowadzeń układu 82501

Nazwa sygnału	Kierunek	Znaczenie
TxC	wy	zegar dla danych nadawanych
TEN	we	zezwolenie na nadawanie
TxD	we	dane nadawane
RxC	wy	zegar danych odbieranych
CRS	wy	wykryto nośny
RxD	wy	dane odebrane
CDT	wy	wykryto kolizję
LPBK/ WDTD	we	wprowadzenie wewnętrznej pętli testowej
TRMT	wy	różnicowy sygnał danych nadawanych do transcivera
RCV	we	różnicowy sygnał danych odbieranych z transcivera
CLSN	we	różnicowy sygnał wykrytej kolizji przez układ transcivera
C1,C2	we	wejścia kondensatora dla układu PLL
X1,X2	we	wejścia oscylatora kwarcowego 20 MHz

Połączenie sterownik - transciver jest najważniejszym poziomem zgodności dla sprzętu różnych producentów spełniającego normę Ethernet. Połączenie to przedstawiono na rysunku 1 (sygnały T2 ÷ T13).

Układ 82501 współpracuje z koprocesorem sieci 82586. Podstawowe funkcje tego układu to: kodowanie i dekodowanie nadawanych bądź odbieranych ramek w kodzie Manchester, zapewnienie dopasowania różnicowych sygnałów do i z transcivera, gene-

racja sygnałów zegarowych dla nadawania, jak również wypracowania poprzez układ PLL sygnału zegarowego dla danych odbieranych.

Układ zawiera także zegar blokujący nadawanie danych, gdy trwało dłużej niż 25 milisekund, na przykład w razie pojawienia się ciągłych zakłóceń, które zostały zinterpretowane jako aktywność na liniach danych. Dla celów testowych element 82501 pozwala zawrócić wewnętrznie dane nadawane do toru odbiornika. W tabeli 2 podano opis wyprowadzeń układu 82501.

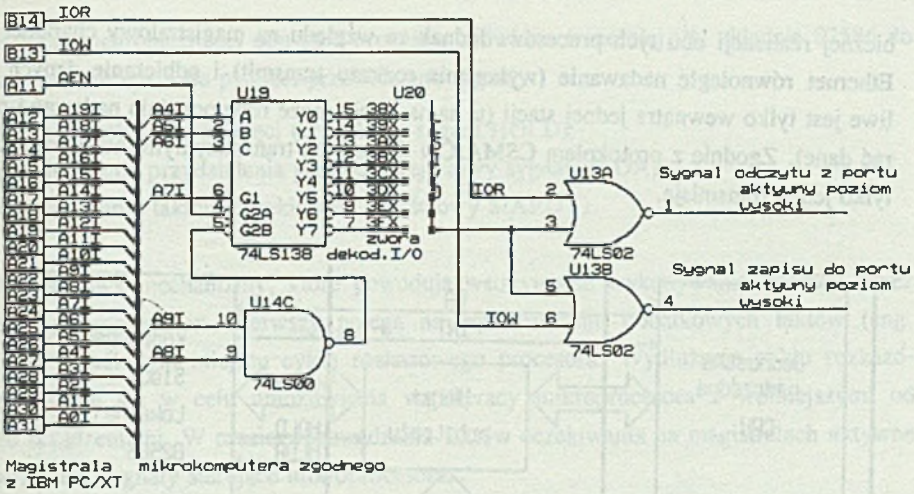
Najważniejszym i najbardziej skomplikowanym elementem sterownika sieci jest koprocesor 82586.

W dalszej części artykułu zostaną przedstawione pewne cechy koprocesora sieci lokalnej oraz wynikające z nich rozwiązania współpracy sterownika sieci lokalnej z jednostką centralną mikrokomputera zgodnego z IBM PC.

3. Budowa sterowników sieci lokalnej z koprocesorem 82586

Patrząc z pewnego punktu widzenia można powiedzieć, że architektura mikrokomputera składa się z jednostki centralnej, pamięci operacyjnej i urządzeń wejścia /wyjścia. Układy takie, jak klawiatura, sprzęg szeregowy RS232, zegar czasu astronomicznego, sterownik graficzny, sterownik dysków, czy sterownik sieci lokalnej są właśnie typowymi urządzeniami wejścia/wyjścia. Istnieje wiele specjalizowanych układów scalonych realizujących funkcje związane z połączeniem urządzeń zewnętrznych. Jednostka centralna posiada osobną przestrzeń adresową przeznaczoną do komunikacji z tymi urządzeniami. Połączenie z jednostką centralną wymaga zaprojektowania pewnego układu kombinacyjnego do wypracowania, na podstawie wystawionych na magistralę adresów i sygnałów sterujących, sygnału wyboru. Na rysunku 2 przedstawiono przykładowy dekoder adresów z przestrzeni urządzeń we/wy, dla mikrokomputera zgodnego z IBM PC.

Urządzenie wejścia/wyjścia zwykle zajmuje kilka adresów w przestrzeni adresowej, przy czym pewne adresy służą do przesyłu danych, inne zaś do sterowania pracą układu scalonego. Pojedynczy rozróżnialny bajt w przestrzeni adresowej urządzeń we/wy często nazywa się za literaturą anglojęzyczną portem. Dla przykładu, układ 8255 w przestrzeni adresowej zajmuje 4 adresy: 3 zajmują porty do przesyłu danych, a pod adresem najstarszym dostępny jest rejestr statusu służący do zaprogramowania trybu pracy pozostałych trzech portów. Dla bardziej skomplikowanych urządzeń liczba przesyłanych informacji sterujących zwykle przekracza jeden bajt.

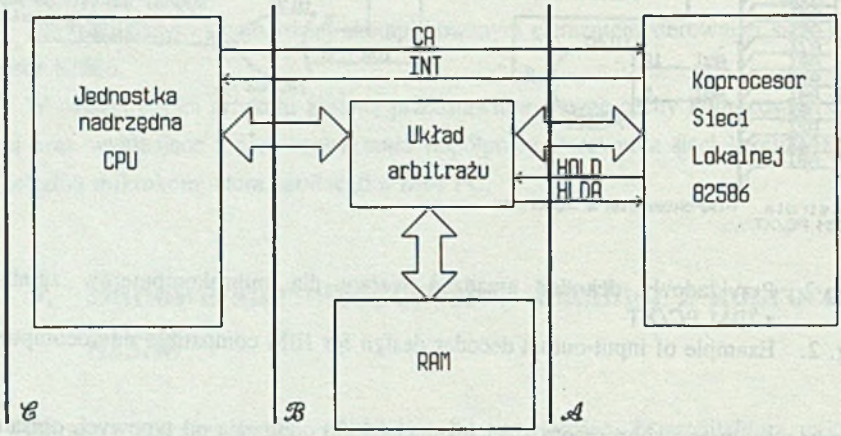


Rys. 2. Przykładowy dekoder urządzeń we/wy dla mikrokomputerów zgodnych z IBM PC/XT

Fig. 2. Example of input-output decoder design for IBM compatible microcomputer

Zasady współpracy koprocesora sieci lokalnej 82586 odbiegają od typowych opisanych wyżej sposobów komunikacji z układem we/wy. Sam koprocesor bardziej przypomina sposobem programowania mikroprocesor niż programowalny układ peryferyjny. Komunikacja pomiędzy koprocesorem a jednostką centralną, którą dalej nazywać będziemy jednostką nadrzędną, odbywa się poprzez dwudostępną pamięć. W pamięci tej przechowywane są wysyłane bądź odebrane z sieci informacje oraz rozkazy dla koprocesora. Jednostka nadrzędna przygotowuje listę rozkazów dynamicznie wraz z odpowiednimi skojarzonymi strukturami danych do wysłania przez sieć, a potem inicjuje działanie koprocesora sygnałem CA (ang. Channel Attention). Od tej chwili układ 82586 działa współbieżnie, to znaczy wykonuje aż do zakończenia całą zadaną mu listę rozkazów. Jeżeli w tym czasie nie nastąpią równocześnie odwołania do wspólnej pamięci, to oba procesy: proces wykonywania operacji przez koprocesor i proces przetwarzania informacji przez jednostkę nadrzędną, mogą realizować się w pełni równolegle. Należy zauważyć, że koprocesor przygotowany jest również do wykonywania swoich dwóch wewnętrznych procesów w tym samym czasie. Żądanie odebrania ramki może nastąpić asynchronicznie w stosunku do wykonywania rozkazów przez koprocesor. Oczywiście, dane z sieci muszą zostać odebrane niezależnie od tego, czy koprocesor wykonuje aktualnie rozkazy, czy też ich nie wykonuje. Z tego też powodu projektanci układu 82586 przewidzieli możliwość wspól-

bieżnej realizacji obu tych procesów. Jednak ze względu na magistralowy charakter sieci Ethernet równolegle nadawanie (wykonanie rozkazu transmit) i odbieranie danych możliwe jest tylko wewnątrz jednej stacji (ta sama stacja może równocześnie nadawać i odbierać dane). Zgodnie z protokołem CSMA/CD w medium transmisyjnym może odbywać się tylko jedna transmisja.



Rys. 3. Schemat blokowy komunikacji pomiędzy koprocesorem 82586 a procesorem nadrzędnym
Fig. 3. Exchanging information scheme between the coprocessor 82586 and the supervisor processor

Na rysunku 3 przedstawiono schemat blokowy ilustrujący sposób komunikacji pomiędzy jednostką nadrzędną a koprocesorem. Układ arbitrażu rozwiązuje konflikt i w przypadku gdyby jednostka nadrzędna i koprocesor w tym samym czasie żądały dostępu do wspólnej pamięci, przyznaje pamięć tylko jednemu urządzeniu.

Wymiana informacji między tymi urządzeniami odbywa się poprzez dwudostępną pamięć. Typowy cykl pracy wygląda następująco: jednostka nadrzędna zadaje odpowiednie rozkazy dla koprocesora, po czym uaktywnia go (sygnałem CA), po wykonaniu rozkazów koprocesor generuje przerwanie (sygnałem INT) i w ten sposób przekazuje sterowanie z powrotem do jednostki nadrzędnej. Najczęściej wykonywany rozkaz (Transmit) wysyła dane w sieć, inne rozkazy służą do celów diagnostycznych lub do ustalenia parametrów pracy stacji. Okresy aktywności jednostki nadrzędnej i koprocesora są rozłączne i zmieniają się naprzemiennie. Konflikt w dostępie do pamięci może wystąpić jedynie wtedy, gdy podczas aktywności jednostki nadrzędnej do stacji przesyłane są przez sieci

dane. O asynchroniczności obu tych procesów mówiono wcześniej. W układzie 82586 do sterowania dostępem do pamięci przewidziano sygnały:

- żądania dostępu do pamięci (wyjściowy sygnał HOLD)
- potwierdzenia przydzielenia pamięci (wejściowy sygnał HLDA)
- wprowadzania taktów oczekiwania (wejściowy S/ARDY).

Istnieją dwa mechanizmy, które powodują wstrzymanie wykonywania operacji przez typowy mikroprocesor. Pierwszy polega na wprowadzeniu dodatkowych taktów (ang. wait) w określonym miejscu cyklu rozkazowego procesora. Wydłużenie cyklu rozkazowego stosuje się w celu umożliwienia współpracy mikroprocesora z wolniejszymi od niego urządzeniami. W czasie wprowadzania taktów oczekiwania na magistralach aktywne są wszystkie sygnały sterujące mikroprocesora.

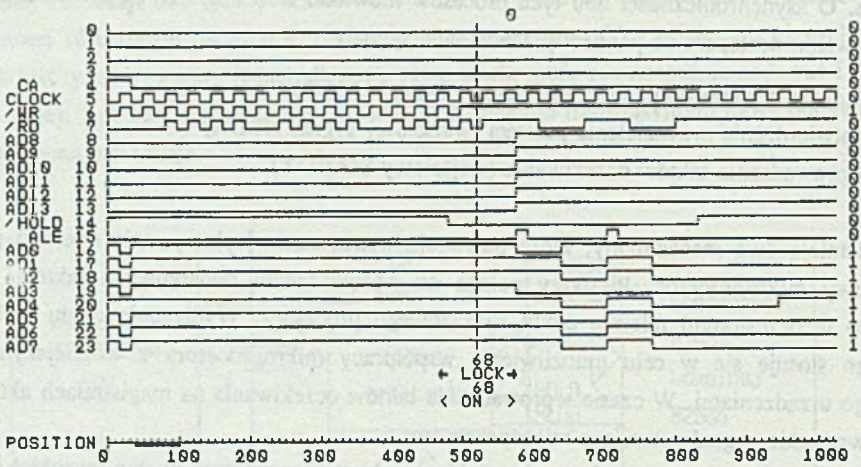
Drugi mechanizm pozwala wprowadzić sygnały mikroprocesora w stan wysokiej impedancji. Po dokończeniu cyklu rozkazowego mikroprocesor przestaje wykonywać program i zwalnia magistralę dla innych układów, które korzystają zwykle z metody bezpośredniego dostępu do pamięci (DMA). Sygnał HOLD jest żądaniem układu 82586 dostępu do pamięci. Koprocesor jednak nie rozpocznie żadnej operacji z pamięcią, póki nie otrzyma potwierdzenia zwolnienia magistrali przez jednostkę nadrzędną na linii HLDA.

Sygnał S/ARDY służy do wprowadzania taktów oczekiwania w czasie dostępu do pamięci, przy czym sygnał ten może być synchronizowany zewnętrznie lub wewnętrznie (stąd też nazwa sygnału ang. Synchronous and Asynchronous Ready). O trybie synchronizacji decyduje aktualnie wybrana konfiguracja układu 82586.

Na rysunku 4 przedstawiono przebiegi zarejestrowane analizatorem stanów logicznych, wybranych sygnałów sterujących układu 82586 w czasie dostępu do pamięci.

Aby przystąpić do projektowania sterownika sieci lokalnej, należy rozwiązać metodę współdzielenia pamięci, co w zasadzie sprowadza się do pytania, gdzie przeprowadzić granicę pomiędzy sterownikiem a mikrokomputerem. Możliwe są trzy rozwiązania.

Linia podziału może prowadzić pomiędzy pamięcią a koprocesorem tak, jak linia A na rysunku 3. Wtedy pamięcią dwudostępną jest pamięć mikrokomputera. W czasie dostępu koprocesora do takiej pamięci niemożliwe jest wykonywanie jakichkolwiek rozkazów przez jednostkę centralną. Do zorganizowania przesyłu informacji można wykorzystać mechanizm bezpośredniego dostępu do pamięci. Zwykle układ DMA przesyła dane pomiędzy pamięcią a portem. Niestety, opierając się na tego typu mechanizmie przesłu danych,nie można podłączyć układu 82586.



Rys. 4. Pobranie dwóch bajtów z pamięci w czasie inicjalizacji koprocesora 82586. Sygnał nr 5 (clock) jest sygnałem zegarowym. Przebiegi zarejestrowane zostały analizatorem stanów logicznych

Fig. 4. Loading 2 bytes from memory during initialization process. Signal number 5 is a clock. Timing are stored using logic analyzer

Koprocesor powinien sam zapisywać bądź pobierać dane z pamięci. Nie jest możliwe zorganizowanie komunikacji koprocesora z pamięcią poprzez port, do którego będą wysyłane lub z którego będą pobierane bloki danych. Układ DMA musi mieć możliwość przesyłu typu pamięć - pamięć, aby koprocesor mógł zaadresować komórki, z których chce skorzystać. Jeżeli pamięć operacyjna opiera się na dynamicznych układach scalonych, to w takiej sytuacji musimy bardzo dokładnie przeanalizować współpracę DMA z układem odświeżania pamięci. Prawidłowe zaprojektowanie układu może okazać się bardzo trudne lub wręcz niemożliwe. Dla przykładu, w mikrokomputerze zgodnym z IBM PC jedyny kanał DMA do przesyłu informacji typu pamięć - pamięć realizuje właśnie odświeżanie pamięci. Dokumentacja serwisowa zaleca, aby ze względu na odświeżanie pamięci, nie wydłużać okresów oczekiwań (wait) powyżej 2,1 mikrosekund (dziesięć taktów zegarowych przy standardowym zegarze 4,77MHz). Układ 82586 po zakończeniu przyjmowania ramki przez cały czas trwania tak zwanej przerwy międzyramkowej rezerwuje sobie dostęp do pamięci. Czas przerwy międzyramkowej jest parametrem, który może zostać programowo ustalony. Typowo zgodnie z normą sieci Etheret wynosi on 9,6 mikrosekund. Jest to czas pięciokrotnie dłuższy niż pożądany ze względu na wstrzymanie odświeżania pamięci. Jak widać, rozwiązanie to, które dalej będziemy nazywać połączeniem

sterownika sieci lokalnej poprzez DMA, jest w dużym stopniu zależnie od konstrukcji mikrokomputera.

Zamiast wykorzystywać kanał DMA, można kartę sterownika sieci wyposażyć we własną pamięć. Na rysunku 3 konstrukcja tego rodzaju schematycznie została zaznaczona przez linię B. Dalej taką konstrukcję będziemy nazywać sterownikiem sieci lokalnej opartym na dwudostępnej pamięci, gdzie jednostką nadrzędną jest procesor mikrokomputera, który korzysta z pamięci znajdującej się na karcie sterownika. Możliwe jest wtedy w pełni współbieżne wykonywanie zadań, np.: odbieranie ramek przez koprocesor i realizowanie procesu obliczeniowego przez procesor mikrokomputera, oczywiście pod warunkiem, że oba układy nie będą chciały w tym samym czasie skorzystać z pamięci na karcie sterownika. Dostęp do wspólnej pamięci musi kontrolować układ arbitrażu, przy czym wyższy priorytet podczas rozwiązywania konfliktów powinien mieć koprocesor.

Poszczególne szeregowo przesyłane bity informacji w sieci Ethernet są przesyłane z odstępem 100 ns. W czasie 800 ns przesłany jest jeden bajt informacji. W koprocesorze istnieją specjalne kolejki FIFO umożliwiające buforowanie informacji pomiędzy jej postacią równoległą w pamięci a szeregową w sieci. W przypadku odbioru niepotrzebne jest, po każdym skompletowaniu bajtu z postaci szeregowej, odwoływanie się do pamięci (lub odwrotnie w przypadku zapisu). Jednakże pojemność buforów FIFO jest ograniczona do 16 bajtów, dlatego układ arbitrażu nie może pozwolić na to, aby dane, które przychodzą lub muszą być wysłane w sieć, nie zostały we właściwym czasie odczytane z pamięci.

Wymaganie zapewnienia odpowiedniej szybkości dla przesyłu danych pomiędzy koprocesorem a pamięcią dotyczy zarówno sterowników opartych na dwudostępnej pamięci, jak i na przesyłie DMA. Zwykle przesył DMA jest szybszy niż dostęp procesora do pamięci. Jednak możliwa jest sytuacja odwrotna; na przykład w niektórych modelach mikrokomputerów zgodnych z IBM PC/AT ze względu na zwiększenie częstotliwości zegara procesora oraz 16-bitową organizację szyny danych, komunikacja z pamięcią poprzez procesor jest szybsza niż poprzez DMA.

Na rysunku 3 linia C odpowiadałaby sterownikowi, który posiada nie tylko własną pamięć, ale również własny procesor. Procesor na karcie jako nadrzędny w stosunku do układu 82586 ma za zadanie wykonywać niższe warstwy oprogramowania sieciowego, a w szczególności protokół połączeniowy. Tego rodzaju rozwiązanie stanowi "nową generację" sprzętu sieciowego (sterownik sieci automatycznie realizujący funkcje wyknywane do tej pory przez oprogramowanie sieciowe, obciążające procesor mikrokomputera).

Brak aktualnie dobrze zdefiniowanego standardu wyższych warstw sieciowych może być przyczyną nieefektywnego wykorzystania tego rodzaju sterownika ze względu na zdublowanie pewnych funkcji. Może zdarzyć się również sytuacja niepotrzebnego kopiowania obszarów danych pomiędzy pamięcią operacyjną a pamięcią sterownika sieci, podczas gdy wystarczające byłoby operowanie przez oprogramowanie sieciowe komputera bezpośrenio na danych znajdujących się w pamięci sterownika.

Niecelowe jest również zastosowanie tego rodzaju sterownika do mikrokomputerów pracujących pod systemem jednoużytkowym takim jak MS DOS. Po odwołaniu się do zasobów sieciowych przez program aplikacyjny formułowane żądanie przekazywane jest sterownikowi sieci. W czasie gdy procesor mikrokomputera czeka w jakiejś programowej pętli na wyniki, procesor na karcie wykonuje oprogramowanie niższych warstw sieciowych związane z obsługą odwołania programu aplikacyjnego, przy czym podczas samych transmisji w sieci aktywnym układem na karcie sterownika będzie koprocesor. W przypadku omawianego przykładu systemu jednozadaniowego można zauważyć, że procesor na karcie sterownika jest zbędny, gdyż procesor mikrokomputera może sformułować żądanie transmisji i zainicjować układ 82586, zamiast czekać w martwej pętli na wyniki.

Zastosowanie procesora na karcie sterownika jest opłacalne dopiero wtedy, gdy system jest wielozadaniowy (np.: system operacyjny zarządzający Serverem sieci Netware), to znaczy kiedy procesor mikrokomputera zamiast czekać w pętli na wyniki pewnych odwołań, przechodzi do wykonania innego zadania.

4. Klasyfikacja konstrukcji sterowników sieci lokalnej

W literaturze wprowadzono podział sterowników sieci lokalnej na inteligentne, czyli takie, które posiadają własny procesor i nieinteligentne - bez procesora. Podział ten nie wydaje się być ścisły, gdyż dla kart bez procesora nazwa nie podaje, w jaki sposób komunikuje się koprocesor z pamięcią, nie wiadomo, czy sterownik nieiteligentny jest oparty na przesyle kanałem DMA czy wykorzystuje pamięć dwudostępną. Ponieważ oparcie sterownika nieinteligentnego na dwudostępnej pamięci ma wiele zalet w stosunku do przesyłu kanałem DMA, można założyć, że mówiąc o sterowniku nieinteligentym będziemy mieli w większości przypadków do czynienia ze sterownikiem z dwudostępną pamięcią.

Podobne niedopowiedzenie kryje się w nazwie sterownik inteligentny. Można przyjąć, że sterownik z własnym procesorem z punktu widzenia architektury maszyn cyfrowych niewiele różni się od koprocesora sieci lokalnej. Oba układy autonomicznie wykonują programy, przy czym różnią się tylko "poziomem" realizowanych protokołów sieciowych

(zgodnie z modelem OSI/ISO). Pozostanie kwestią otwartą, w jaki sposób sterownik inteligentny komunikować się będzie z procesorem mikrokomputera. Istnieje tu pełna analogia z sterownikiem nieinteligentnym.

Są trzy możliwe sposoby rozwiązania takiej komunikacji: opierając się na rejestrach umieszczonych w przestrzeni adresowej urządzeń wejścia/wyjścia, na kanale DMA i wykorzystując dwudostępną pamięć RAM.

Sterownik sieci lokalnej jest urządzeniem złożonym, dlatego nie wydaje się praktycznie uzasadnione zorganizowanie komunikacji wyłącznie opierając się na portach wejścia/wyjścia. Jednakże ze względu na to, że możliwość taka teoretycznie istnieje, została tutaj wymieniona. Pozostałe dwa typy komunikacji przez DMA i pamięć dwudostępną są analogiczne do omówionych wcześniej dla sterownika nieinteligentnego, przy czym należy zauważyć, że są one bardziej "równoprawne". Łatwiejsze w wypadku sterownika inteligentnego jest zorganizowanie komunikacji typu pamięć - port, przez co można ominąć wiele ograniczeń konstrukcyjnych. W konsekwencji rozwiązanie oparte na dwudostępnej pamięci RAM nie ma tak znacznej przewagi, jak to było w wypadku sterownika nieinteligentnego. Dla sterowników sieci lokalnej opartych na układzie 82586 można wprowadzić następujące podziały:

a/ ze względu na umieszczenie procesora nadrzędnego:

- sterowniki inteligentne - SI,
- sterowniki nieinteligentne - SN,

b/ ze względu na sposób komunikacji:

- sterowniki z dwudostępną pamięcią RAM DPRAM
- sterowniki z kanałem DMA - DMA.

Możemy zatem mówić o następujących sterownikach sieci lokalnej:SN DMA, SN DPRAM, SI DMA, SI DPRAM, SI DMA + DPRAM.

Jako teoretyczną możliwość należałoby również wymienić sterownik SI, to znaczy taki, który komunikuje się z procesorem mikrokomputera wyłącznie za pomocą portów w przestrzeni adresowej urządzeń we/wy.

5. Podsumowanie

Omawiany podział ze względu na sposób rozwiązania konstrukcji sterownika sieci lokalnej powstał w wyniku prac badawczych prowadzonych na Politechnice Śląskiej. Ich efektem jest opracowanie oryginalnej konstrukcji sprzętowej dla magistralowej sieci E-

thernet. Opracowany sterownik sieci aktualnie współpracujący z oprogramowaniem Netware firmy Novell jest konstrukcją SN DPRAM.

Szczegółowe informacje dotyczące konstrukcji i podziału sterowników sieci lokalnej mogą być przydatne nie tylko wąskiemu gronu konstruktorów tego rodzaju sprzętu. Mogą być one bardzo pomocne dla użytkowników, którzy mają do rozwiązania problem skompletowania odpowiedniego zestawu sieciowego. Wiadomo, że o efektywności całej instalacji sieci decyduje (tak jak w łańcuchu) jej najsłabsze ogniwo. Najwolniejszy element całej instalacji ogranicza jej przepustowość. Istotne zatem są pewne szczegóły konstrukcyjne sterowników sieciowych, gdyż ich poznanie i analiza pozwala na optymalne skonfigurowanie zestawu sieciowego.

LITERATURA

[1] Local Area Networking (LAN) Component User's Manual, Intel 1985.
[2] P.Strutyński.: Przegląd sieci lokalnych dla IBM PC. Informatyka, 11, 12 1988.
[3] P.Dembiński.: Protokoły komunikacyjne część 1, 2. Informatyka 5, 6 1990.
[4] The Ethernet-ALocal Network. Version 1.0 Digital, Intel, Xerox, 1990.
[5] Dokumentacja prototypowych rozwiązań sieci lokalnej. Praca zbiorowa CPBR 8.7 Cel 15, Gliwice 1989.
[6] Materiały z konferencji "Mikrokomputery w sieci lokalnej". Gdańsk-Sopot 1989.

Recenzent: Doc. dr hab. inż. Adam Mrózek

Wpłynęło do Redakcji 7 października 1992 r.

Abstract

There are many computer input-output function which you can support using a coprocessor. Local area network controller based on the coprocessor is not an exception.

Intel is one of the world leader in designing integrated circuits field. In many cases the solution of Intel was world standard's. It may be interesting to know how Intel's Local Area Networks Coprocessor 82586 works in more general point of view not only networks applications.

In this paper I considered various ways of designing local area networks controller based on 82586 coprocessor. The main problem that you can resolve is the information exchange between the LAN coprocessor and the microcomputer main processor. You can use I/O ports and to ports memory. As a shared memory it is possible to use microcomputer main or on controller board memory. The controller may be intelligent or not intelligent that means on controller board there is general propose processor or it is not.

In the end I discuss the classification of local area networks controllers.