

mikroKLAN 6

P. 1877/84

6

1984

# informatyka

Systemy mikrokomputerowe  
Samotest  
Fotony w komputerze?

Nr 6

Miesięcznik Rok XIX

Czerwiec 1984

Organ Komitetu Informatyki  
MNSzWiT oraz Komitetu  
Naukowo-Technicznego NOT  
ds. Informatyki

**KOLEGIUM REDAKCYJNE:**

Mgr inż. Zbigniew GLUZA, mgr Teresa JABLONSKA (sekretarz), Władysław KLEPACZ (zastępca redaktora naczelnego), prof. dr hab. Leon LUKASZEWICZ (redaktor naczelnny), mgr inż. Andrzej J. PIOTROWSKI, mgr Andrzej SZALAS, dr inż. Janusz ZALEWSKI

**STALE WSPÓLPRACUJĄ:**

Mgr Adam B. EMPACHER, dr Janusz GWIAZDA (Libia), mgr Katarzyna ISAK, dr Jacek OWCZARCZYK, mgr Marek SOBCZYK, dr Jakub TATAR-KIEWICZ, mgr inż. Teresa WILCZEK

**PRZEWODNICZĄCY  
RADY PROGRAMOWEJ:**

Prof. dr hab. Tadeusz PECHE

Materialów nie zamówionych redakcja nie zwraca

Redakcja: 00-041 Warszawa, ul. Jasna 14/16, pok. 243 i 244, tel. 27-71-40 lub 26-82-61 w. 184

Zakł. Graf. „Tamka”. Zam. 2110. Obj. 4,0 ark. druk. Nakład 4000 egz. T-41.

INDEKS 36124

Cena egzemplarza zł 75,—  
Prenumerata roczna zł 900,—

WYDAWNICTWO  
MACZELNA ORGANIZACJA TECHNICZNA  
CZASOPISMI / NSZĄZEK TECHNICZNYCH  
  
SIGMA

**W NUMERZE:**

	Strona
Charakterystyka jednostrukturalnych mikrokomputerów 8-bitowych na przykładzie INTELA 8048 <i>Marek Rakowski, Andrzej T. Rosiński</i>	2
PROG-2 — uniwersalny programator pamięci stałych <i>Marek Pawłowski, Andrzej Woźniak</i>	6
Mikrokomputerowy system modułowy VME <i>Krzysztof Rzymkowski</i>	9
MULTIBUS-II — wieloprocesorowy system modułowy <i>Roman Trehciński</i>	21
<b>mikroKLAN</b>	13
— NEPTUN 184	
— VisiCalc	
— 8255 w praktycznych zastosowaniach	
— Pamięci pseudostatyczne	
— Sinclair QL	
— O jabłku opowieść (1)	
<b>SAMOTESTY</b>	24
— I/A. Struktura i modularność programów	
<b>Z KRAJU</b>	25
— Posiedzenie Rady ds. Zastosowań ŚTO	
<b>ZE ŚWIATA</b>	26
— Fotony w komputerze	
— Przemysł informatyczny 1983 w ocenie brytyjskiej	
<b>RECENZJE</b>	30
— Elementarz mikroprocesorowy	
<b>TERMINOLOGIA</b>	31
— O najczęstszych błędach w terminologii mikrokomputerowej (1)	
CZWARTA OKŁADKA — <i>Jacek Gawłowski</i>	

**W NAJBLIŻSZYCH NUMERACH:**

- Roman Żelazny o narzędziach inżynierii oprogramowania
- Jerzy Dworzecki o bibliotece oprogramowania CPM
- Ryszard Rybus o oprogramowaniu systemu mikroprocesorowego bez pamięci masowej
- Zbigniew Poznański i Jerzy Dańda o języku PL/M
- Andrzej Szalas o systemach ekspertowych
- Ewa Gutman o bazie danych, obsługiwanej przez komputer pośredniczący
- Jerzy Szyller o mikroprocesorach lat osiemdziesiątych
- Józef de Mezer o sterowaniu alfaskopem

# Technika mikroprocesorowa — rok później

Jakie zmiany zaszły w rozwoju techniki mikroprocesorowej w Polsce od ukazania się ubiegłorocznych numerów *INFORMATYKI*, poświęconych w całości tej tematyce? Okazuje się, że półtora roku to okres zbyt krótki, aby nastąpił jakiś zasadniczy zwrot w tej dziedzinie, a nawet — by można zaobserwować większe zmiany.

Rozpoczęto produkcję systemów wspomaganie RTDS i MSWP przedstawionych w numerze 1/1983, opisywanych także w innych czasopismach (*Biuletyn MERA*, nr 4/82, 5/83), i wydaje się, że na ten temat już nie można powiedzieć wiele nowego. Należy jedynie oczekiwać, że wraz z rozpowszechnieniem tych systemów zaczną do nas docierać sygnały o ich jakości i funkcjonalności. Pewną pozytywną niespodzianką jest ewolucja programowanej stacji przygotowania danych PSPD-90 w kierunku zestawu uniwersalnego, służącego przede wszystkim do wspomaganie projektowania systemów mikroprocesorowych. Podjęcie prac w tym kierunku wynika zapewne z ciągłego braku narzędzi dla projektantów systemów mikroprocesorowych. Z tym większym zainteresowaniem należy śledzić prace nad rozwojem systemu uruchomieniowego EMU-48, przeznaczonego dla polskich mikrokomputerów jednokładowych. Choć jeszcze nie ma ich na rynku, być może niebawem usłyszymy o podjęciu krajowej produkcji mikrokomputerów 4- lub 8-bitowych — na razie musimy zadowolić się ich opisem. Będą to układy o niewielkich możliwościach obliczeniowych, przeznaczone głównie do zastosowań w sprzęcie powszechnego użytku oraz prostych zastosowań profesjonalnych (por. PAK, nr 1, 1984). Dlatego producent mikroukładów angażuje się w większym stopniu w konstruowanie systemów uruchomieniowych.

Ogólnie, lista narzędzi potrzebnych projektantom jest dość długa i obejmuje środki zarówno programowe jak i sprzętowe. Jednym z podstawowych urządzeń, które musi posiadać każdy zespół projektowy, jest programator pamięci PROM. Bardzo pożądaną cechą programatora jest jego uniwersalność — choć trudno dostosować się do parametrów wszystkich pamięci, które są lub ukażą się jeszcze na rynku. Brak krajowej produkcji programatorów zmusza projektantów do indywidualnego konstruowania takich urządzeń. Jest to sytuacja nie do pomyślenia w krajach o wysoko rozwiniętej technice. Jeżeli zespół projektowy potrzebuje narzędzi, to prawie nigdy nie zajmuje się ich konstruowaniem, gdyż natychmiast lub z niewielkim opóźnieniem pojawiają się one na rynku i można je kupić. U nas sytuacja jest odwrotna.

Taka też droga prowadziła do zbudowania programatora PROG-2, niezbędnego w procesie rozwoju systemu MSM — obecnie dobrze wyposażonego w narzędzia od strony sprzętowej. Współtwórcy tego systemu (nr 2, 1983) mogliby z pewnością przedstawić o wiele więcej doświadczeń zdobytych przy jego konstruowaniu, co zresztą czynią przy innych okazjach (konferencje krajowe i międzynarodowe), dając świadectwo właściwego podejścia do zagadnień projektowania systemów mikroprocesorowych. Niestety, nie ma zbyt wielu nowych rozwiązań, a przynajmniej nie udało mi się ich zauważyć. Przykładowo, własnym życiem — jakby niezależnie od rozwoju techniki — żyją 16-bitowe mikrokomputery MERA-60. Sądzę, że przydałaby się im miniaturyzacja, gdyż obecnie sprawiają wrażenie minikomputerów i z biegiem czasu mogą stać się mniej konkurencyjne. Spośród innych zestawów omówionych w numerze 2, 1983 wyróżnia się system MIKROSTER (Systemy Mikroprocesorowe, nr 12, 1983), którego produkcję rozpoczęto w zakładach UNITRA-CEMI w Szczytnie (PAK, nr 3, 1984). Jeżeli okaże się on dostatecznie uniwersalny, a przede wszystkim niezawodny, to może stanowić poważną konkurencję dla innych systemów modułowych. Bezwładność naszego przemysłu odbiera nadzieję na zmianę tego stanu.

Twórców przedstawionej w numerze 2/1983 aparatury camacowskiej nie interesują natomiast rozwiązania polowiczne, tzn. oparte na magistrali małoprzepustowej lub odbiegającej od norm światowych, i będą oni zmierzać prawdopodobnie w kierunku konstrukcji bardziej uniwersalnej. Z tym większą uwagą warto przeczytać w bieżącym numerze artykuły dotyczące dwóch najnowocześniejszych systemów modułowych — magistrali VME i MULTIBUS-II. Oba standardy umożliwiają konstruowanie systemów w pełni wieloprocessorowych w jednej kasie, a także —

łączenie wielu kaset. Oba nadają się też do tworzenia zarówno małych systemów 8-bitowych, jak i dużych wieloprocessorowych zestawów opartych na mikroprocesorach 16- lub 32-bitowych. Dopiero w tym drugim przypadku ujawniają się, zresztą, ich prawdziwe zalety. Nie są to bowiem jedynie zwykłe systemy sprzęgające, lecz — znormalizowane zestawy przeznaczone do tworzenia bardzo złożonych jednostek obliczeniowych, a więc w Polsce mające znaczenie raczej przyszłościowe. Nie należy jednak zapominać, że w tej dziedzinie przyszłość u nas to terazniejszość, a nie raz już przeszłość na Zachodzie.

Tam trwa prawdziwa „magistralowa” wojna, której nie można było zauważyć jeszcze przed rokiem. Firmy INTEL i MOTOROLA w walce o rynek prześcigają się w tworzeniu coraz doskonalszych technicznie systemów. Specyfikacja VME ma niewiele ponad 3 lata, a już obecnie bloki tego systemu (ok. 200 typów) wytwarza ok. 60 producentów zrzeszonych w niedawno powołanej organizacji VME Bus Manufacturers Group. Choć jest to w zasadzie system ukierunkowany na mikroprocesor MOTOROLA 68000, istnieją także sterowniki oparte na innych mikroprocesorach 16-bitowych, jak np. Z8000 i NS 16032. Z kolei firma INTEL, zdając sobie sprawę ze stopniowego starzenia się systemu MULTIBUS-I (mimo istnienia ok. 1000 różnych typów modułów produkowanych przez 170 firm), poświęciła dwa lata na opracowanie systemu dostosowanego do nowocześniejszych wymagań (niezgodności między MULTIBUS-I i MULTIBUS-II rozwiązują się tworząc zestawy hybrydowe). Kilka innych firm amerykańskich, jak FOXBORO, NCR, PRIME, TEKTRONIX, a także zachodniemiecki SIEMENS, natychmiast zadeklarowało chęć współdziałania w realizacji tego standardu. Jest to więc sprawa dużej wagi i szybkie przyjęcie odpowiedniej normy krajowej mogłoby być bardzo korzystne (por. artykuły na temat sprzęgów w numerach 1, 5, 1983).

W krajach zachodnich, kwestia wyboru między VME i MULTIBUS-II wiąże się z dostępnością wyrobów — choć MULTIBUS-II jest nowocześniejszy, bloki standardu VME są już produkowane. W Polsce, przy braku modułów w obu przypadkach może zadecydować większa nowoczesność magistrali MULTIBUS-II oraz jej zgodność z wyrobami firmy INTEL.

Oczywiście, przymusowa unifikacja nie ma wiele sensu, jednak przy wyborze zakresu produkcji sprzętu komputerowego warto kierować się rozsądkiem. Nie powinno być wątpliwości, że znacznie łatwiej i taniej jest produkować w każdym z zakładów od jednego do kilku typów specjalizowanych bloków według jednolitego standardu, niż w każdym z tych zakładów rozwijać produkcję pełnego systemu, a więc całego zakresu bloków według własnych standardów. Aby oszacować skutki braku normalizacji, wystarczy pomnożyć średni koszt opracowania jednego bloku przez liczbę standardów. Oczywiście, korzyści z normalizacji nie polegają jedynie na uniknięciu niepotrzebnych opracowań, standardowa produkcja ułatwia bowiem rozszerzenie asortymentu i zwiększenie produkcji wyrobów. Kwestia uzgodnienia krajowego standardu pozostaje tu jednak nadal otwarta.

Co widać na mikroprocesorowej mapie „rok później”? Z pewnością ekspansja tej techniki jest znaczna, czego wyrazem choćby powstanie na naszych łamach mikroKLANU. Ale w praktyce nadal brakuje elementów i narzędzi, choć obserwuje się tu stopniową poprawę sytuacji. Prowadzi się wiele prac rozwojowych, czego dowodem jest coraz większa liczba artykułów publikowanych w czasopismach i w materiałach konferencyjnych. Nadal jednak mało wiadomo o zastosowaniach mikroprocesorów, nie ma ich w naszym otoczeniu. Nadal też nie zauważa się postępu w dziedzinie systemów wieloprocessorowych. Okazuje się jednak, że w krajowych warunkach nie jest łatwo skonstruować i oprogramować zestaw wielomikroprocesorowy użyteczny w praktyce. Z tego względu, a także ze względu na potrzeby normalizacji prostszych zestawów, coraz pilniejszy staje się problem wyboru krajowego standardu konstrukcyjnego. Przedstawiony tu obraz techniki mikroprocesorowej w Polsce zostanie rozszerzony w następnym numerze o zagadnienia oprogramowania.

# Charakterystyka jednostrukturalnych mikrokomputerów 8-bitowych na przykładzie INTELA 8048

Pojawienie się mikrokomputerów jednostrukturalnych znacznie poszerzyło dziedzinę zastosowań układów scalonych. Zawierają one bowiem w jednym układzie wszystkie elementy występujące w systemach mikroprocesorowych, m.in. pamięć danych i pamięć programu, dzięki czemu mogą pracować samodzielnie, bez układów dodatkowych, jak pamięci RAM, ROM czy sterownik systemowy. Właściwości te umożliwiają tanią, jednoukładową realizację części sterujących urządzeń.

Duża uniwersalność produkowanych obecnie układów powoduje, że pojęcie mikrokomputera jednostrukturalnego nie jest jednoznaczne (por. [5], Vol. 1). Niektóre układy mogą wykonywać program zapisany w wewnętrznej pamięci stałej, jak i przetwarzać dane zgodnie z instrukcjami pobieranymi z zewnątrz. Podobnie operacje wejścia-wyjścia mogą być realizowane przez porty zawarte wewnątrz układu, jak i przez porty zewnętrzne. Ten sam układ bywa zatem określony jako mikrokomputer, jeżeli pracuje samodzielnie, lub jako mikroprocesor, jeżeli jest częścią składową bardziej rozbudowanych systemów [7]. Poniżej pod pojęciem mikrokomputera rozumiane będą układy klasy mikroprocesorów zawierające pamięć stałą (pamięć programu) i pamięć o dostępie swobodnym (pamięć danych).

## CHARAKTERYSTYKA MIKROKOMPUTERÓW JEDNO-STRUKTURALNYCH

Spśród wielu jednostrukturalnych mikrokomputerów 8-bitowych, opracowanych do tej pory, na uwagę zasługują m.in. układy serii 8048 (8048, 8748, 8049, 8021, 8022) firmy INTEL [4, 5], serii 1650 (1650, 1655, 1670, 1645) firmy GENERAL INSTRUMENTS [5], serii 3870 [5] i 3872 [1] firmy MOSTEK oraz układ Z8 firmy ZILOG [1, 7]. Listy rozkazów tych układów obejmują rozkazy dodawania, dodawania z przeniesieniem (Z8 i układy serii 8048), odejmowania (Z8 oraz układy serii 3870 i 1650), operacji logicznych, przesłań (Z8 również przesłań blokowych), przesunięcia cyklicznego skoku do podprogramu i powrotu z podprogramu, skoków bezwarunkowych i warunkowych.

Cechą tych układów jest także obecność wewnętrznej pamięci danych i wewnętrznej pamięci programu. Pojemności tych pamięci są różne, w zależności od przeznaczenia — np. układ typu 1645 zawiera wewnętrzną pamięć danych o pojemności 16 bajtów, natomiast pojemność wewnętrznej pamięci danych układów typu 3872, 8049 i Z8

wynosi 128 bajtów. Podobnie układ typu 1645 posiada wewnętrzną pamięć programu o pojemności 256 słów, a układ 3872 — 4 K słów. Warto zaznaczyć, że układy serii 1650 mają 12-bitowe słowo, a wszystkie rozkazy układów tej rodziny są jednobajtowe. W niektórych przypadkach istnieje możliwość rozbudowy pamięci danych i programu w oparciu o układy zewnętrzne. Takie właściwości mają układy serii 8048 (pamięć programu może być rozbudowana do 4 K słów, a pamięć danych o 256 bajty) oraz układ Z8 (zarówno pamięć danych, jak i programu może być rozbudowana o 62 K bajty). W układach tych poszczególne rozkazy lub dane są pobierane z pamięci wewnętrznej, jeżeli ich adresy nie przekraczają pojemności tej pamięci, lub — z pamięci zewnętrznej w przypadku przeciwnym.

Mikrokomputery jednostrukturalne mają układy odmierzenia czasu (ang. timer) i układy rejestracji zdarzeń (liczniki zdarzeń). Układy serii 1650 mają jeden licznik zdarzeń, nie generujący przerwań, natomiast układy typu 8048 i 3870 — jeden układ czasowy — licznik zdarzeń oraz tzw. prescaler (w układach 3870 ma on programowaną pojemność). Mikrokomputer Z8 ma dwa układy czasowe — liczniki zdarzeń i dwa układy typu prescaler o programowanej pojemności.

Parametrami mikrokomputerów jednostrukturalnych są także: możliwość generowania przerwań (w układach typu 1650 i 1655 przerwanie nie są możliwe, natomiast w układzie Z8 możliwych jest sześć różnych przerwań), liczba wejść i wyjść, wielkość stosu, możliwość konwersji analogowo-cyfrowej (układ typu 8022), możliwość szeregowej transmisji danych (układ Z8), itp.

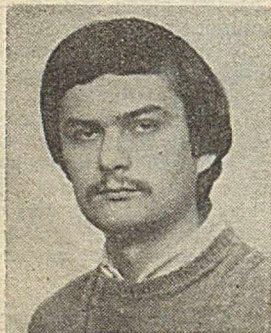
Szczegółowe właściwości funkcjonalne mikrokomputerów jednostrukturalnych zostaną omówione na przykładzie układu INTEL 8048.

## ARCHITEKTURA UKŁADU INTEL 8048

Na rysunku 1 przedstawiono schemat blokowy mikrokomputera INTEL 8048 [6] (INTEL 8035 różni się od niego jedynie brakiem pamięci ROM). Wyróżniono na nim następujące bloki funkcjonalne:

- sterowanie centralne I
- rejestr rozkazów z dekoderelem
- sterowanie centralne II
- rejestry T/C i PC z inkrementerem<sup>1)</sup>

<sup>1)</sup> T/C — ang. timer/counter, układ czasowy — licznik zdarzeń, PC — ang. program counter, licznik rozkazów



Mgr inż. MAREK RAKOWSKI ukończył w 1979 r. Wydział Automatyki i Informatyki Politechniki Śląskiej. Pracuje w Instytucie Technologii Elektronowej przy NPCP. Zajmuje się strukturą logiczną układów LSI MOS.



Dr inż. ANDRZEJ ROSIŃSKI ukończył w 1968 r. Wydział Elektroniki Politechniki Warszawskiej. Pracuje w Instytucie Technologii Elektronowej. Początkowo zajmował się wykorzystaniem komputerów do projektowania tranzystorów oraz cyfrowych układów scalonych małej skali integracji. Jest współautorem pierwszego opracowanego w kraju mikroprocesora. Obecnie pracuje nad strukturą logiczną mikrokomputera 8-bitowego.

- pamięć programu z rejestrami adresowymi
- pamięć danych z układem adresowania
- arytmometr
- porty we-wy.

Poszczególne przrutniki rejestru słowa stanu programu PSW (przeniesienie CY, przeniesienie pomocnicze AC, bit flagowy FO, wybór zestawu rejestrów RB i 3-bitowy licznik stosu) umieszczono w różnych miejscach schematu, w zależności od tego, z którymi podzespołami są związane.

### Sterowanie centralne I

Sterowanie centralne I obejmuje bufor wszystkich sygnałów wejściowych i wyjściowych — poza portami. W bloku tym znajduje się układ oscylatora, współpracujący z zewnętrznym układem RC lub kryształem kwarcu. Na podstawie przebiegu wyjściowego oscylatora o danym okresie generowane są w sposób ciągły dwa sygnały ( $\Phi 1$ ,  $\Phi 2$ ) o czasie trwania równym temu okresowi, oddalone o połowę okresu. Zatem na jeden takt wewnętrzny składają się trzy okresy zegara. Takty wewnętrzne, oznaczone dalej P1-P5, trwają pomiędzy kolejnymi narastającymi zboczami pierwszego sygnału. Pięć taktów wewnętrznych tworzy cykl maszynowy. Wszystkie rozkazy są jedno- lub dwucyklowe, a więc cykl rozkazowy trwa piętnaście lub trzydzieści okresów oscylatora.

W sterowaniu centralnym I znajduje się 5-bitowy pre-skaler (modulo 32) zliczający cykle maszynowe, gdy włączony jest układ czasowy zliczający przepelnienia.

### Rejestr rozkazów z dekodere

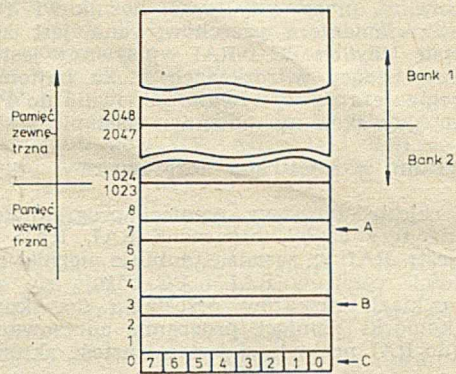
W rejestrze rozkazów (RI) przechowywane są kody wykonywanych rozkazów. Zapis do RI następuje w pierwszym taktie wewnętrznym cyklu rozkazowego. Na podstawie zawartości RI w dekodery, dekodowane są pojedyncze rozkazy lub ich grupy.

### Sterowanie centralne II

Sterowanie centralne II, wraz ze sterowaniem portami i sterowaniem arytmometrem, wytwarza sygnały sterujące pracą poszczególnych podukładów na podstawie stanu wyjść dekodera rozkazów a także — sygnałów sterowania centralnego I oraz zdekodowanych taktów wewnętrznych P1-P5.

W układzie możliwe są dwa rodzaje przerwań — zewnętrzne (zgłaszane na wejściu INT) i pochodzące od układu czasowego licznika. Po przyjęciu zgłoszenia

przerwania, sterowanie centralne II powoduje wykonanie rozkazu skoku do podprogramu, do adresu 003(H) lub 007(H) (rys. 2).



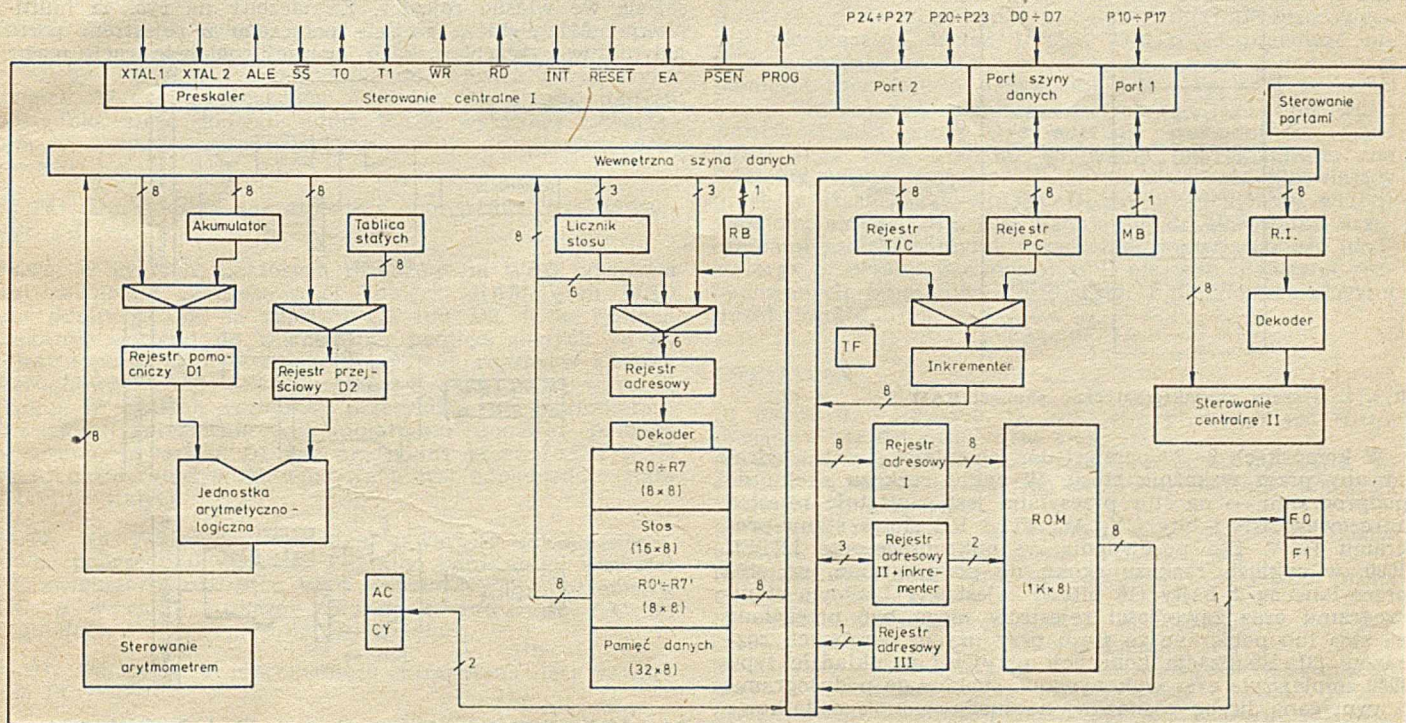
Rys. 2. Obszary charakterystyczne pamięci programu  
A) komórka, do której wykonywany jest skok ze śladem po przerwaniu od T/C  
B) komórka, do której wykonywany jest skok ze śladem po przerwaniu zewnętrznym  
C) komórka, od której rozpoczyna się wykonywanie programu po zaniku sygnału RESET

Podczas pracy krokowej, gdy działanie układu jest zatrzymane niskim poziomem napięcia na wejściu SS (np. 1) sterowanie centralne II wpisuje do RI kod rozkazu NO OPERATION.

### Rejestry T/C i PC z inkrementerem

Układ czasowy — licznik zdarzeń i dolny bajt licznika rozkazów korzystają ze wspólnego inkrementera. Inkrementer ma własny rejestr, do którego wpisywana jest inkrementowana wartość.

Układ czasowy — licznik zdarzeń może być wyłączony lub — pracować w trybie odmierzenia czasu albo zliczania zdarzeń, polegających na zmianie sygnału wejściowego T1 z logicznej jedynki na zero. Zawartość układu jest wstępnie ustawiana rozkazem przesłania zawartości akumulatora. Przepelnienie T/C powoduje wygenerowanie przerwania (jeżeli odpowiednie przerwanie jest dozwolone) i ustawienie przrutnika TF. Przrutnik ten jest zerowany podczas rozkazu skoku warunkowego, przy warunku TF=1.



Rys. 1. Schemat blokowy mikrokomputera INTEL 8048

W rozkazach jedno-cyklowych oraz w pierwszych cyklach maszynowych rozkazów dwucyklowych T/C i PC korzystają z inkrementera na przemian. W drugich cyklach maszynowych rozkazów dwucyklowych inkrementer jest przyporządkowany przez cały czas licznikowi rozkazów. W rejestrze inkrementera przechowywana jest zawartość PC — w czasie, gdy do PC i RAI wpisywany jest kolejny bajt programu, będący w rozkazach skoku bajtem adresu. Dane z inkrementera są wpisywane ponownie do PC i RAI w przypadku, gdy skok nie powinien nastąpić.

### Pamięć programu z rejestrami adresowymi

12-bitowy rejestr adresowy pamięci programu RA jest podzielony na trzy części, oznaczone RAI, RAI I i RAI II (rys. 1). Rejestr RAI w zasadzie dubluje licznik rozkazów PC. Zawartości rejestrów RAI i PC różnią się od siebie jedynie w cyklach rozkazów przesłania do akumulatora zawartości komórki pamięci programu adresowanej akumulatorem (do RAI przesyłana jest zawartość akumulatora).

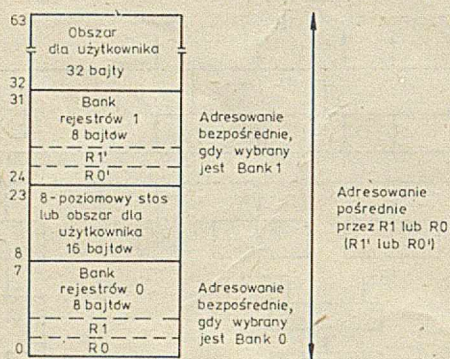
Część RAI I rejestru RA posiada własny 3-bitowy inkrementer. Zawartość RAI I jest inkrementowana podczas przepełnień PC.

Najbardziej znaczący bit adresu pamięci programu jest przechowywany w RAI II. Jest to bit wyboru banku pamięci, modyfikowany wyłącznie w rozkazach skoku, kiedy do RAI II przepisywany jest stan przerzutnika MB (ang. memory bank). Przerzutnik MB jest zerowany i ustawiany oddzielnymi rozkazami. W czasie wykonywania podprogramu obsługi przerwania rozkazy skoku powodują wyzerowanie RAI II, tj. skok w obrębie banku 0.

Wszystkie rozkazy układu INTEL 8048 są jedno- lub dwubajtowe. Bajty pamięci programu o adresie mniejszym lub równym 1023 są pobierane z pamięci wewnętrznej, a bajty o adresie większym — z pamięci zewnętrznej (rys. 2). Mikroprocesor typu 8035 pobiera wszystkie rozkazy z zewnątrz.

### Pamięć danych z układem adresowania

Wewnętrzna pamięć danych układu (rys. 3) zawiera 64 komórki, które mogą być adresowane pośrednio sześcioma dolnymi bitami komórki o adresie 0 lub 1 albo — jeżeli przerzutnik wyboru zestawu rejestrów RB jest ustawiony — komórki 24 lub 25. Ponadto komórki o adresach 0—7 (rejestry R0—R7) albo — jeżeli RB jest ustawiony — komórki 24—31 (rejestry R0—R7) mogą być adresowane bezpośrednio.



Rys. 3. Obszary charakterystyczne pamięci RAM

W komórkach 8—23 pamięci danych mieści się stos, adresowany przez wskaźnik stosu. W cyklu rozkazu skoku do podprogramu — na stos przesyłana jest zawartość rejestru adresowego RA i bity CY, AC, FO i RB słowa stanu programu PSW. Dla porównania, w mikroprocesorze INTEL 8080, w cyklach rozkazu skoku do podprogramu, na stos przesyłane są 2 bajty (16 bitów) rejestru PC. Słowo stanu programu oraz zawartości rejestrów mogą być przesyłane na stos lub pobierane ze stosu przy użyciu odrębnych rozkazów [3]. Realizacja koncepcji przyjętej w układzie typu 8048 zmniejszyła czas wykonywania skoków do podprogramu i wymaganą liczbę rozkazów, narzuciła jednak dodatkowe ograniczenia — przy założeniu, że ślad zapamiętywany jest w dwóch komórkach, zapamiętywana część rejestru PSW nie może przekraczać 4 bitów.

Stos układu INTEL 8048 pozwala na zagnieżdżanie podprogramów (w tym podprogramu obsługi przerwania) do ośmiu poziomów.

Jeżeli system oparty na układzie typu 8048 zawiera dodatkową pamięć o dostępie swobodnym, zewnętrzną w stosunku do mikrokomputera, to komunikacja z tą pamięcią odbywa się za pomocą dwóch oddzielnych rozkazów przesłania zawartości akumulatora do pamięci zewnętrznej i z pamięci zewnętrznej do akumulatora. Zewnętrzna pamięć danych adresowana jest pośrednio zawartością rejestrów R0 lub R0' lub R1' (jeżeli przerzutnik RB jest ustawiony).

### Arytmometr

W skład arytmometru wchodzi, oprócz jednostki arytmetyczno-logicznej (JAL), trzy rejestry (akumulator, współpracujący z akumulatorem rejestr pomocniczy D1, rejestr przejściowy D2) i tablica stałych, o wymiarze 16x8 bitów. W tablicy znajdują się następujące stałe [6]:

- 00<sub>(H)</sub> — stała wykorzystywana m.in. podczas przesyłania zawartości akumulatora na wewnętrzną szynę danych
- 00<sub>(H)</sub>, 06<sub>(H)</sub>, 60<sub>(H)</sub>, 66<sub>(H)</sub> — stałe wykorzystywane podczas korekcji dziesiętnej zawartości akumulatora
- FF<sub>(H)</sub> — stała wykorzystywana w cyklach rozkazu dekrementacji
- 01<sub>(H)</sub> — stała wykorzystywana w cyklach rozkazu inkrementacji oraz podczas wyboru warunku skoku
- 02<sub>(H)</sub>, 04<sub>(H)</sub>, 08<sub>(H)</sub>, 10<sub>(H)</sub>, 20<sub>(H)</sub>, 40<sub>(H)</sub>, 80<sub>(H)</sub> — stałe wykorzystywane podczas wyboru warunku skoku.

Zawartości rejestru pomocniczego D1 i przejściowego D2 są poddawane w JAL jednej z następujących operacji:

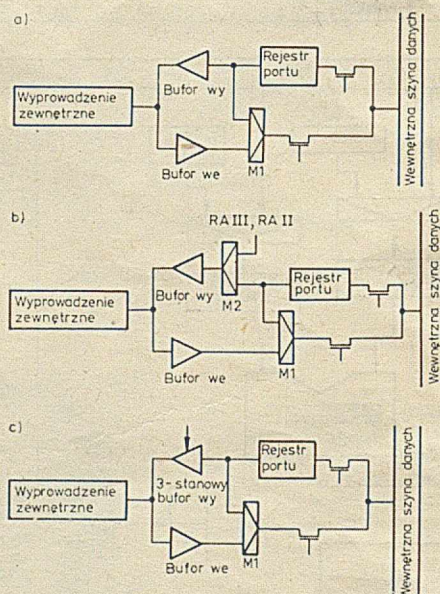
- dodawania
- dodawania z przeniesieniem
- przesunięcia cyklicznego w prawo (rejestr D1)
- sumy logicznej
- iloczynu logicznego
- sumy modulo 2.

Przesunięcie cykliczne w lewo zawartości akumulatora jest wykonywane przez przesłanie zawartości akumulatora do rejestrów D1 oraz D2 i dodanie D1 do D2.

W cyklu rozkazu skoku do podprogramu, w arytmometrze zapamiętywana jest również przed przesłaniem na stos (i jeżeli zachodzi potrzeba — inkrementowana) zawartość rejestru RAI.

### Porty wejścia-wyjścia

Mikrokomputer ma trzy 8-bitowe porty (rys. 4), wyprowadzone we własne rejestry. Zaznaczony na rys. 4a multiplexer M1 wybiera wejście połączone z rejestrem portu jedynie w cyklach rozkazu operacji na rejestrach portu.



Rys. 4. Konfiguracja portów: a) port P1 i bardziej znacząca tetrada portu P2, b) mniej znacząca tetrada portu P2, c) port szyny danych

Wyjścia portu P1 są sterowane zawartością rejestru portu. Rozkaz wczytania do akumulatora danych z portu P1 powoduje w rzeczywistości wczytanie iloczynu logicznego zawartości rejestru portu P1 i danych podawanych z zewnątrz.

Główna tetrada<sup>2)</sup> portu P2 ma taką samą strukturę jak w porcie P1. Tetrada dolna służy dodatkowo do wysłania na zewnątrz najbardziej znaczących bitów adresu pamięci programu podczas pracy z zewnętrzną pamięcią programu. Multiplexer M2 (rys. 4b) wybierze wówczas na przemian najbardziej znaczącą tetradę rejestru adresowego pamięci programu i najmniej znaczącą tetradę rejestru portu P2. Mniej znacząca tetrada portu P2 pełni dodatkowo funkcję komunikacji z ekspanderem we-wy typu 8243. Komunikacja odbywa się w cyklach rozkazów związanych z ekspanderem przy mniej znaczące bity rejestru portu.

Bufor wyjściowy portu szyny danych jest trójstanowy. W zależności od konfiguracji systemu, port szyny danych może zawierać w sposób ciągły dane zapisane w rejestrze portu bądź też pracować jako trójstanowy, w pełni dwukierunkowy port we-wy. W trybie pracy z zewnętrzną pamięcią programu przez port szyny danych odbywa się przesłanie mniej znaczącego bajtu adresu i kodu rozkazu. W cyklach rozkazów dostępu do zewnętrznej pamięci danych przez port szyny danych odbywa się przesłanie adresu pamięci i danych.

## WYBRANE MIKROOPERACJE

Większość przesłań w układzie odbywa się przez wewnętrzną szynę danych WSD. Podobnie jak w mikroprocesorze 8080 [2] w fazie  $\Phi 1$  WSD przyjmuje stan  $FF_{(H)}$ , a przesłania przez WSD następują w fazach  $\Phi 2$  i polegają na rozładowaniu odpowiednich linii tej szyny. Poniżej podano wybrane mikrooperacje oraz chwile ich wykonywania dla rozkazów jedno- oraz dwucyklowych.

### Rozkazy jednocyklowe oraz pierwszy cykl maszynowy rozkazów dwucyklowych

**P1. $\Phi 2$ .** Na WSD jest wysyłany kod rozkazu z pamięci ROM, portu szyny danych lub sterowania centralnego II. Stan WSD jest wpisywany do RI, do rejestru D1 wpisywana jest zawartość akumulatora, a do rejestru D2 stała  $00_{(H)}$ . Do poszczególnych przerzutów rejestru adresowego pamięci RAM wpisuje się odpowiednio 0, RB i trzy najmniej znaczące bity kodu rozkazu, równe  $00r$  dla rozkazów adresujących pamięć RAM pośrednio lub  $rrr$  dla rozkazów adresujących pamięć RAM bezpośrednio (wartości  $r$  są określone przez użytkownika).

**P2. $\Phi 2$ .** Na WSD wysyłany jest kolejny adres pamięci programu z inkrementera, a stan WSD jest przepisywany do rejestrów PC RAI i ewentualnie do rejestru portu szyny danych. W cyklach rozkazów adresujących pamięć danych pośrednio, sześć dolnych bitów RAM, wybranej zawartością rejestru adresowego, jest wpisywana do rejestru adresowego.

**P3. $\Phi 01$ .** Do rejestru inkrementera wpisywana jest zawartość T/C.

**P3. $\Phi 2$ .** W cyklach niektórych rozkazów na WSD wysyłana jest zawartość zaadresowanej komórki RAM, stan WSD jest przepisywany do rejestru D1 lub D2, a w cyklach rozkazów dostępu do zewnętrznej pamięci danych — do rejestru portu szyny danych. W cyklach rozkazów skoków warunkowych — do D1 przez WSD wpisywane są warunki skoku. W cyklach rozkazów przesłania do akumulatora zawartości adresowanej akumulatorem komórki pamięci programu, zawartość D1 jest przesyłana przez JAL i WSD do RAI. Do rejestru D2 wpisywana jest odpowiednia stała z tablicy stałych.

**P4. $\Phi 2$ .** Na WSD jest wysyłana zawartość inkrementera, równa poprzedniej wartości T/C albo poprzedniej wartości T/C zwiększonej o 1. Stan WSD jest wpisywany do rejestru T/C, a w cyklu rozkazów przesłania zawartość T/C do akumulatora — również do akumulatora.

**P5. $\Phi 1$ .** Do rejestru inkrementera wpisywana jest zawartość PC.

**P5. $\Phi 2$ .** W cyklach niektórych rozkazów na WSD wysyłane są dane z JAL. Stan WSD jest przepisywany do akumulatora, a w cyklach rozkazów wyjścia — do rejestru odpowiedniego portu. W cyklach rozkazów działań na rejestrach portów, zawartość odpowiedniego rejestru portu jest przesyłana przez WSD do rejestru D1. W cyklu rozkazów skoku do podprogramu — do rejestru D1 przez WSD są przesyłane bity CY, AC, FO i RB oraz zawartości RAIII i RAI. W cyklach rozkazów powrotu z podprogramu następuje przesłanie przez WSD do RAIII, RAI (ewentualnie — przerzutników CY, AC, FD i RB) — zawartości adresowanej wskaźnikiem stosu komórki RAM.

### Drugi cykl maszynowy rozkazów dwucyklowych

**P1. $\Phi 2$ .** W cyklach rozkazów działań na rejestrach portów, zawartość rejestru odpowiedniego portu jest przesyłana przez WSD do rejestru D2. W cyklach rozkazów powrotu z podprogramu na WSD wysyłana jest zawartość komórki pamięci RAM adresowanej wskaźnikiem stosu, a w cyklach pozostałych rozkazów dwucyklowych — zawartość adresowanej komórki ROM albo portu szyny danych. Stan WSD jest wpisywany do rejestrów D2, PC, RAI, ewentualnie do rejestru portu szyny danych, a w cyklu rozkazów przesłania danych bezpośrednich do rejestru — do adresowanej komórki RAM.

**P2. $\Phi 2$ .** W cyklach rozkazów wejścia stan odpowiedniego portu jest przesyłany przez WSD do akumulatora. W cyklach rozkazów skoku i skoku do podprogramu, poprzez dolną połowę WSD, następuje przesłanie zawartości MB do RAIII i bitów  $A_{10}$ ,  $A_9$ ,  $A_8$  z RI do RAI.

**P3. $\Phi 2$ .** W cyklach niektórych rozkazów dane z inkrementera są przesyłane przez WSD do rejestrów PC, RAI, ewentualnie — do rejestru portu szyny danych. W cyklu rozkazów skoku do podprogramu dane z inkrementera są przesyłane przez WSD na stos.

**P4. $\Phi 2$ .** Na WSD wystawiane są dane z JAL, a stan WSD może być przepisany do akumulatora. W cyklu rozkazów skoku do podprogramu stan WSD przesyłany jest na stos.

**P5. $\Phi 1$ .** Do rejestru inkrementera wpisywana jest zawartość PC.

\* \* \*

Z przedstawionej charakterystyki mikrokomputerów jednostrukturalnych na przykładzie układu INTEL 8048 wynika, że ich architektura, a zwłaszcza pamięć programu i danych, system przerwań oraz organizacja i wielkość stosu decydują o tym, iż nadają się one do zastosowań prostych, gdzie stosowanie systemów mikroprocesorowych jest nieopłacalne. Układ INTEL 8048 może pracować nie tylko samodzielnie, ale również w systemach z dodatkową pamięcią programu o pojemności 3 K słów oraz z dodatkową 256-bajtową pamięcią danych, a jego możliwości wejścia-wyjścia mogą być znacznie rozbudowane przez układ typu 8243. Ponadto omawiany mikrokomputer ma 8-bitowy układ czasowy — licznik zdarzeń, ułatwiający pracę w czasie rzeczywistym. Dzięki wymienionym właściwościom, układy typu 8048 nadają się do sterowania urządzeniami profesjonalnymi, urządzeniami powszechnego użytku oraz procesami przemysłowymi, nie wymagającymi rozbudowanych programów sterujących i nie dostarczającymi dużej ilości danych.

## LITERATURA

- [1] Capece R. P., Posa J. G., eds.: Microprocessors and Microcomputers. Mc Graw-Hill, New York, 1981
- [2] Chachulski M., Krzyżanowski Z., Kunciewicz W., Rosiński A. T.: Architektura mikroprocesora typu 8080. Prace Instytutu Technologii Elektronowej, z. 10, 1980
- [3] Instytut Technologii Elektronowej. Układ 8-bitowego mikroprocesora MCY 7880N. ZOINTE ITE, Warszawa, 1983
- [4] INTEL Corp.: MCS-48TM Microcomputer users manual. Santa Clara (California), 1976
- [5] Osborne A.: An introduction to microcomputers. Vol. 1-2. Osborne and Associates, Berkeley (California), 1978
- [6] Rakowski M., Rosiński A. T.: Mikrokomputer jednostrukturalny — architektura, układy logiczne, zasada działania. Opracowanie wewnętrzne, Instytut Technologii Elektronowej, Warszawa, 1983
- [7] ZILOG Inc.: Z8 MCU Microcomputer Product Specification. Preliminary, Cupertino (California), 1979.

<sup>2)</sup> Tetrada, słowo rzadko używane w informatyce, oznacza u autorków 4 bity (przyp. red.)

# PROG-2

## uniwersalny programator pamięci stałych

Od kilku lat wzrasta w kraju zainteresowanie pamięciami stałymi programowanymi elektrycznie — PROM. Jest to efekt uniwersalności tego rodzaju układów. Pamięci stałe stosuje się bowiem jako:

- układy przechowujące stałe oprogramowanie systemów mikroprocesorowych,
- pamięci mikroprogramu w systemach mikroprogramowych,
- generatory znaków,
- uniwersalne układy kombinacyjne, zastępujące całe fragmenty sieci logicznej, dotychczas tworzone z pojedynczych bramek.

Zastosowanie pamięci stałych nie tylko upraszcza proces projektowania, ale prowadzi do minimalizacji liczby układów. Podstawową zaletą układów z taką pamięcią jest możliwość modyfikacji funkcji bez zmiany połączeń — jedynie przez zmianę zawartości pamięci. Zalety te mogły być jednak w pełni wykorzystane dopiero po pojawieniu się pamięci programowanych bezpośrednio przez użytkownika (PROM). Zapisywanie informacji do pamięci stałych wymaga podania impulsów programujących o specyficznych parametrach elektrycznych i czasowych.

Produkowane obecnie pamięci stałe programowane elektrycznie różnią się następującymi cechami:

- technologią (bipolarne — PROM, MOS UVEPROM, EAROM<sup>1)</sup>)
- organizacją (np. 32x8, 256x4, 2Kx8, 8Kx8 bitów itd.)
- zasilaniem (+5V; +5V, +12V, -5V; +5V, -9V itp.)
- typem obudowy (16-, 18-, 20-, 24-, 28-końcówkowe)
- parametrami programowania, tj.:
  - poziomami napięć programujących (np. 48V dla pamięci typu 1702A, 25V — dla 2716, 10,5V — dla 74S188)
  - wartościami natężenia prądu impulsów programujących (5 mA dla pamięci typu 3624, 240 mA — dla TM624)
  - liczbą sygnałów programujących (zwykle jeden dla pamięci MOS, np. 2716, kilka dla pamięci bipolarnych, np. dwa dla TM624)
  - parametrami czasowymi impulsów programujących, takimi, jak: szybkość narastania i opadania (np. 0,4 V/ $\mu$ s dla MM63XX, 5 V/ $\mu$ s dla MM63LSXX), czas trwania (od 200  $\mu$ s dla 36XX do 700 ms dla SN 74188), zależności czasowe między kolejnymi impulsami (pojedynczy impuls dla SN 74188 i 2716, seria identycznych impulsów dla 2708, seria impulsów o wzrastającym czasie trwania dla I36XX).

Zasady programowania, podawane przez producenta, muszą być ściśle przestrzegane. Nie dotrzymanie specyficznych parametrów programowania może bowiem spowodować błędne zaprogramowanie, zmianę zawartości podczas użytkowania lub uszkodzenia pamięci. Ścisłe przestrzeganie procedur programowania jest szczególnie ważne dla pamięci bipolarnych, ponieważ ich programowanie jest procesem nieodwracalnym. Nawet małe odstępstwo od wymagań producenta może spowodować zmniejszenie wydajności programowania (zmniejszenie stosunku liczby prawidłowo zaprogramowanych pamięci do liczby pamięci programowanych) oraz pogorszenie niezawodności pamięci.

Do programowania pamięci stałych używa się specjalnych urządzeń zwanych programatorami. Mogą być one przeznaczone do programowania jednego typu pamięci lub zestawu pamięci o podobnych procedurach programowania.

Przykładami takiego rozwiązania są programatory: Automatic Programmer firmy ELEKTRONIK DIGITALTECHNIK, System 37 firmy CITEL oraz PRISS-20 opracowany w CNPSS MERA-STER w Katowicach, programujące pamięci MOS UVEPROM (2708, 2716 itp.). Istnieje również wiele programatorów budowanych ad hoc przez użytkowników dla jednego typu pamięci.

Rozwój technologii, pojawianie się wciąż nowych pamięci stałych, o coraz większych pojemnościach i krótszych czasach dostępu, powoduje, że co jakiś czas użytkownik staje przed problemem kupna lub konstrukcji nowego programatora. Sytuację komplikuje dodatkowo fakt, że wielu producentów wytwarza pamięci zgodne funkcjonalnie — według obudowy i rozkładu końcówek — różniące się jednak zasadami programowania. Problem ten jest szczególnie uciążliwy w pracowniach konstrukcyjnych i naukowo-badawczych, gdzie przygotowuje się nowe konstrukcje urządzeń cyfrowych. W tej sytuacji celowe jest posiadanie uniwersalnego programatora, który można przystosować do programowania dowolnego typu pamięci.

W programatorach uniwersalnych, część programatora służąca bezpośrednio do obsługi programowanej pamięci jest wymiennalna. W zależności od konstrukcji — elementem wymiennym może być pakiet (zestaw pakietów), jak np. w węgierskim programatorze TR-9562, lub mechanicznie jednolita wkładka, jak w programatorach serii M900 firmy PRO-LOG [2]. Te ostatnie stanowią typowe rozwiązanie konstrukcyjne — istnieje kilka wersji takiego programatora, od najprostszych, jak M910, przeznaczonych do zastosowań przemysłowych, do najbardziej rozbudowanych jak M980, przeznaczonych do laboratoriów badawczych. Wymienne wkładki przeznaczone do programowania określonych typów pamięci są jednolite konstrukcyjnie i mogą być użyte z dowolnym programatorem serii M900<sup>2)</sup>.

Na początku 1978 roku w Instytucie Informatyki PW opracowaliśmy i uruchomiliśmy uniwersalny programator pamięci stałych UPM-2 [1]. Urządzenie to zaprojektowane na układach MSI TTL można szybko przystosować do programowania najnowszych typów pamięci. Jednak niewielka liczba funkcji, a w szczególności mała pojemność pamięci buforowej ogranicza efektywność tego programatora. Dlatego powstała nowa wersja — PROG-2, o konstrukcji zbliżonej do programatora M980 firmy PRO-LOG.

### OPIS FUNKCJONALNY

Programator PROG-2 jest przeznaczony przede wszystkim dla laboratoriów naukowo-badawczych, tzn. dla użytkowników często zmieniających typ programowanych pamięci. Z tego powodu PROG-2 jest samodzielnym urządzeniem mikroprocesorowym, wyposażonym w zestaw funkcji umożliwiających wprowadzanie danych do programowania z klawiatury (umieszczonej na płycie czołowej) lub z dowolnego urządzenia znakowego wyposażonego w sprzęg Jednolitego Systemu, V-24 lub sprzęg dalekopisowy z pętlą prądową 20 mA. Prosta obsługa funkcji oraz czytelna sygnalizacja błędów ułatwia użytkownikowi posługiwanie się programatorem.

PROG-2 realizuje następujące funkcje:  
**DEF** — deklaracja typu programowanej pamięci  
**IN** — wprowadzanie danych do pamięci buforowej z pa-

<sup>1)</sup> PROM — Programmable Read-Only Memory, MOS — Metal-Oxide Semiconductor, UVEPROM — Ultra-Violet Erasable ROM, EAROM — Electrically Alterable ROM

<sup>2)</sup> Cena programatorów serii M900 wynosiła w 1982 roku od 1900 do 5000 dol. za część stałą, a ceny wkładek wynosiły od 500 do 1700 dol. [3]



mięci programowanej (podstawka COPY), z pamięci wzorcowej (podstawka MASTER), ze sprzęgu równoległego JS EMC (styk S-4), ze sprzęgu szeregowego V24 (styk S-2) lub sprzęgu dalekopisowego

**OUT** — wyprowadzanie danych z pamięci buforowej, pamięci programowanej, pamięci wzorcowej, sprzęgu równoległego JS EMC, szeregowego V24 lub dalekopisowego

**VER** — sprawdzanie identyczności danych między dowolnymi, wymienionymi wyżej źródłami danych

**PRG** — programowanie pamięci COPY zawartością pamięci buforowej

**COP** — programowanie pamięci COPY zawartością pamięci wzorcowej MASTER

**ERA** — wymazywanie elektryczne zawartości całej pamięci COPY (dla pamięci typu EEPROM lub EAPROM).

Rozszerzeniem powyższego zestawu są funkcje dodatkowe:

**MOD** — modyfikacja danych w pamięci buforowej, umożliwiająca programowanie pamięci, zanegowaną zawartością lub pod zanegowanymi adresami

**SWP** — wybór mniej lub bardziej znaczącego półbajtu przy operacjach na pamięciach o słowie 4-bitowym

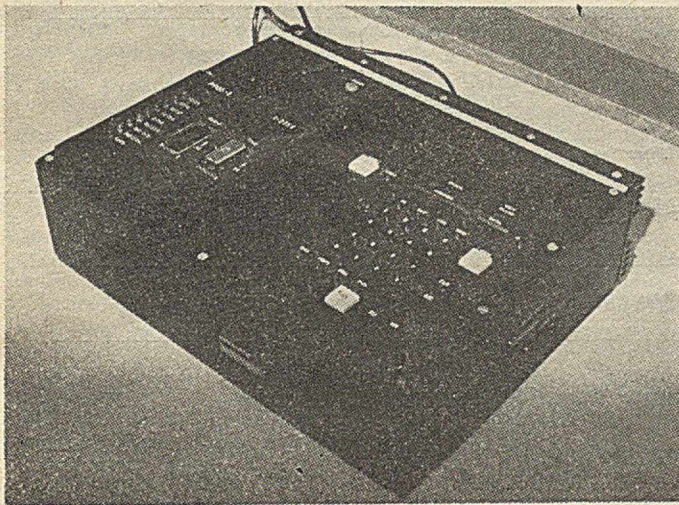
**BLK** — sprawdzenie fabrycznego stanu pamięci (same 0 lub 1)

**PAG** — ustawienie obszaru roboczego w pamięci buforowej

**BAUD** — ustawienie szybkości transmisji szeregowej.

W celu zwiększenia niezawodności działania, PROG-2 został wyposażony w funkcje diagnostyczne i kontrolne, wykonywane automatycznie podczas pracy. Należą do nich:

- kontrola działania podstawowych zespołów programatora bezpośrednio po włączeniu zasilania
- ciągła kontrola działania pamięci buforowej
- kontrola przygotowania wkładki do programowania zadeklarowanego typu pamięci
- kontrola poprawności zasilania pamięci programowanej i wzorcowej.



Fot. 1. Wygląd zewnętrzny programatora PROG-2 z wkładką I27XX

Wybór funkcji jest możliwy przez wciśnięcie odpowiedniego przycisku na płycie czołowej (fot. 1). Potwierdzenie wyboru określonej funkcji jest sygnalizowane za pomocą diody elektroluminescencyjnej. Parametry funkcji wprowadza się za pomocą przycisków numerycznych. Żądanie wykonania funkcji jest wprowadzane za pomocą przycisku EXEC. Wykonywanie funkcji programatora może być przerwane przyciskiem STOP EXEC.

Włożenie wkładki w przeznaczone dla niej miejsce w płycie czołowej powoduje ustawienie programatora w stan początkowy, pozwalający zadeklarować typ programowanej pamięci. Każda wkładka ma pamięć stałą ROM, zawierającą procedury sterujące funkcjami. Programator sprawdza, czy podany typ pamięci może być zaprogramowany przy użyciu określonej wkładki.

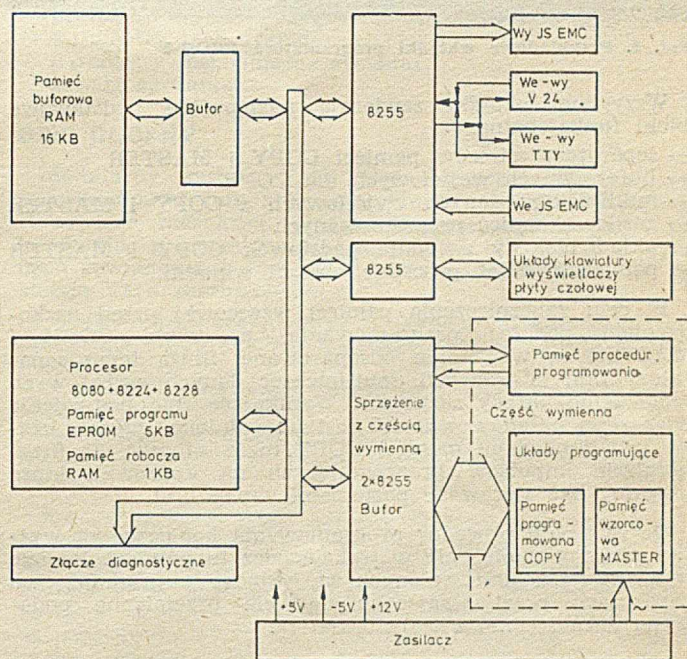
Jeżeli programator nie wykonuje żadnej z wymienionych funkcji to możliwe jest użycie klawiatury współpracującej

z pamięcią buforową. Cyfry szesnastkowe są wyprowadzane na wyświetlacze adresu lub danych, zależnie od stanu diody obok wyświetlacza danych, ustalonego przyciskiem D/A. Odczyt danych z pamięci następuje po wciśnięciu przycisku RD, a zapis — po wciśnięciu WR. Wciśnięcie każdego przycisku jest sygnalizowane dźwiękiem. Dźwiękiem zwraca się również uwagę operatora na sytuacje awaryjne i błędy.

## REALIZACJA UKŁADOWA

W celu realizacji tak bogatego zestawu funkcji, programator PROG-2 wyposażono w system mikroprocesorowy. Schemat blokowy programatora, przedstawiony na rysunku 1, zawiera:

- procesor typu 8080 z układami 8224, 8228, buforami szyny adresowej (4x8216), układami pracy krokowej i przerwań
- dekodery adresów pamięci
- pamięć roboczą RAM 1 KB
- pamięć programu EPROM 6 KB
- układ urządzeń we-wy (8255)
- sprzężenie z częścią wymienną (dwa układy typu 8255)
- pamięć buforową RAM 16 KB
- blok płyty czołowej z układami klawiatury i wyświetlaczami szesnastkowymi
- układ części wymiennej (wkładki) z układami programującymi i pamięcią procedur obsługi wkładki.

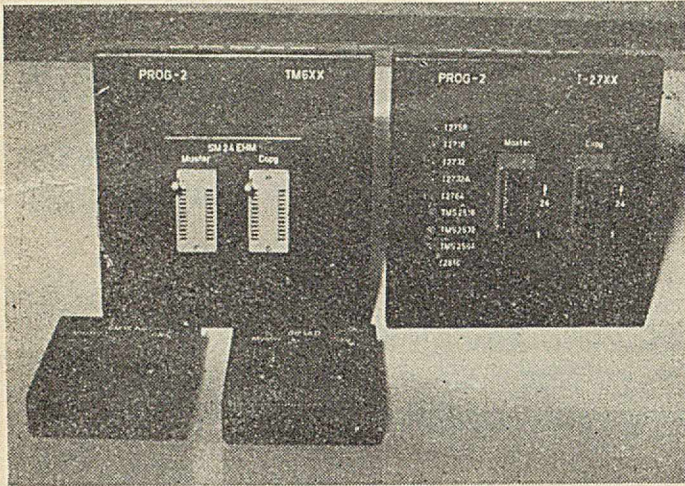


Schemat blokowy programatora PROG-2

Współpraca między częścią stałą programatora a wkładką jest realizowana za pomocą 48 linii dołączonych przez złącza szufladowe ELTRA 50 do dwóch układów typu 8255. Znaczenie sygnałów oraz kierunek ich przesyłania są uzależnione od aktualnie umieszczonej w programatorze wkładki. Ze względów ekonomicznych, a także dla wygody użytkownika — wkładka powinna programować maksymalną liczbę typów pamięci. Ten ostatni warunek jest jednak trudny do spełnienia, dlatego też każda konkretna realizacja jest kompromisem między liczbą programowanych pamięci a skomplikowaniem układów wkładki. W PROG-2 przyjęto dwa podejścia do konstrukcji wkładek (fot. 2).

Jeżeli pamięci mają identyczny typ obudowy, podobne rozmieszczenie sygnałów na końcówkach oraz zbliżone procedury programowania, to na płycie czołowej wkładki umieszczane są bezpośrednio dwie podstawki: dla pamięci programowanej — COPY oraz dla pamięci wzorcowej — MASTER. Wewnątrz wkładki znajduje się zestaw kluczy tranzystorowych dostosowujących wkładkę do określonego typu pamięci. Tak rozwiązana jest wkładka służąca do programowania pamięci UVEPROM. Jeżeli cała rodzina pamięci jest programowana identyczną procedurą, ale poszczególne typy pamięci różnią się pojemnością, organizacją, rozmieszczaniem końcówek i typem obudowy, to na płycie czołowej wkładki umieszczane są dwa złącza szufladowe, do których jest dołączony adapter podstawek (za-

wierający podstawki pod pamięci MASTER i COPY). Tak rozwiązana jest wkładka programująca pamięci bipolarne firmy MONOLITHIC MEMORIES. Dzięki wprowadzeniu jednolitego rozmieszczenia sygnałów na złączach, adaptery podstawek programatora PROG-2 są wspólne dla wielu wkładek.



Fot. 2. Przykładowe wkładki programatora PROG-2

Wszystkie wkładki zawierają następujące podstawowe bloki funkcjonalne:

- separatory adresów pamięci COPY i MASTER
- bufor danych wejściowych dla COPY
- multiplexer danych wyjściowych z COPY i MASTER
- zasilacze napięć programowanych
- układ blokady zasilania podstawek COPY i MASTER
- pamięć EPROM procedur obsługi wkładki.

W celu zabezpieczenia pamięci wzorcowej przed uszkodzeniem podczas programowania, podstawki COPY i MASTER są wzajemnie odseparowane. Służą temu separatory linii adresowych, multiplexer danych wyjściowych i oddzielne układy zasilające. Wynikające stąd podwojenie większości układów wkładki jest uzasadnione, gdyż w czasie programowania pamięci COPY może zdarzyć się przeniknięcie impulsów programujących na końcówki, które powinny być sterowane poziomami napięć TTL.

Do realizacji procedur programowania konieczne są wyższe poziomy napięć niż w trakcie odczytu pamięci. Są one wytwarzane w umieszczonych we wkładce programowanych zasilaczach przetwarzających napięcia przemiennie podawane bezpośrednio z transformatora.

Wkładki są wyposażone w układy blokowania zasilania i impulsów programujących przy zaniku zasilania układów logicznych wkładki bądź rozprogramowania układów typu 8255 (RESET). Chroni to pamięci MASTER i COPY przed uszkodzeniem w przypadku chwilowego zaniku napięcia w sieci. W celu zabezpieczenia pamięci COPY i MASTER przed uszkodzeniami, które mogą powstać np. przez odwrotne włożenie układów scalonych w podstawki, bezpośrednio po wyłączeniu napięć zasilających sprawdzany jest i sygnalizowany stan zasilania pamięci COPY i MASTER.

Procedury obsługi wkładek są przechowywane w pamięci EPROM, umieszczonej wewnątrz każdej z wkładek. Pamięć

ta traktowana jest jako fragment pamięci programu części stałej programatora.

W tabeli przedstawiono zestawienie typów pamięci stałych programowanych za pomocą dotychczas opracowanych wkładek dla programatora PROG-2.

Zestawienie pamięci programowanych za pomocą PROG-2

Oznaczenie wkładki	Przeznaczenie	Typ pamięci
I27XX	Pamięci unipolarne firm INTEL, TEXAS INSTRUMENTS, MOTOROLA	2716, 2732, 27C4, 2758, 2816, 2732A, TMS 2516, TMS 2532, TMS 2564
I87XX	Mikroprocesory z pamięcią EPROM firmy INTEL	8741, 8748, 8755A
I36XX*)	Pamięci bipolarne firmy INTEL	3601, 3621, 3602, 3622, 3605, 3625, 3604, 2624, 3608, 3628, 3616, 3636
TM6XX*)	Pamięci bipolarne firmy TUNGSRAM i MONOLITHIC MEMORIES (63XX-1 i 63XX-2)	TM601, TM622, TM624, 6300, 6301, 6305, 6306, 6308, 6309, 6330, 6331, 6335, 6336, 6340, 6341, 6348, 6349, 6350, 6351, 6352, 6353, 6380, 6381
SN74XXX*)	Pamięci bipolarne firmy TESLA i TEXAS INSTRUMENTS	MH 74188, MH 74S287, SN 74188, SN 74S188, SN 74S288, SN 74S287, SN 74S387, SN 74S470, SN 74S471, SN 74S472, SN 74S473, SN 74S474, SN 74S475, SN 74S476, SN 74S477
I2708	Pamięci unipolarne o zasilaniu +5, +12 i -5V firm INTEL, TEXAS INSTRUMENTS, MOTOROLA	2708, TMS 2716
I1702	Pamięci firmy INTEL	1701, 1702, 1702A
5603A	Pamięci bipolarne serii 56XX firm TUNGSRAM i INTERSIL	5603, 5623, 5604, 5624
K500RXX	Pamięci bipolarne produkcji ZSRR	K565RE4, KR565RT5

\*) Wkładka z adapterami podstawek

#### LITERATURA

- [1] Pawłowski M., Woźniak A.: Uniwersalny programator pamięci stałych UPM-2. Pomiary Automatyka Kontrola, nr 12, 1978
- [2] PRO-LOG Corporation: PROM User's Guide. January 1979
- [3] PRO-LOG Corporation: Personality Module Selection Guide. Price List. May 24, 1982.

## INFORMATYKA za 40 zł! Tylko do końca roku

Od 1 lipca do 31 grudnia 1984 obowiązuje tańsza prenumerata ulgowa, do której mają prawo nowi prenumeratorzy: członkowie indywidualni SNT NOT, studenci, uczniowie szkół zawodowych (zasadniczych, średnich i pomaturalnych).

Cena ulgowa egzemplarza — 40 zł; prenumerata

kwartalna — 120 zł. Zamówienia na IV kwartał można składać do 31 sierpnia.

Przed dokonaniem wpłaty konieczne jest poświadczanie Koła SNT, uczelni lub szkoły — na odcinku przekazu NBP dla adresata — posiadacza rachunku. Na przekazie trzeba podać tytuł czasopisma i okres prenumeraty.

# Mikrokomputerowy system modułarny VME

Zarysowana w latach sześćdziesiątych tendencja do tworzenia dużych uniwersalnych systemów elektronicznych przeznaczonych głównie do automatyzacji pomiarów lub do sterowania procesów technologicznych utrzymuje się do dnia dzisiejszego. Udana próba szerokiego wprowadzenia takich systemów, opracowanych przez różne organizacje międzynarodowe lub firmy produkujące specjalizowaną aparaturę elektroniczną, spowodowały — wraz z rozwojem nowych technologii — pojawienie się dalszych propozycji rozwiązań systemowych uwzględniających możliwości współczesnej elektroniki.

Przykładowo, w latach siedemdziesiątych, w technice jądrowej dominowały systemy NIM<sup>1)</sup> i CAMAC. Obecnie systemy tego rodzaju są w wielu zastosowaniach niewystarczające. Wybór jednego uniwersalnego systemu, spełniającego jednocześnie np. wymagania eksperymentów fizyki wielkich energii jak i wymagania stawiane systemom przetwarzania danych i sterowania bloków energetycznych, mimo wielu podobieństw, nie wydaje się równie łatwy jak w latach sześćdziesiątych.

W powstających propozycjach rozwiązań systemowych, wykorzystuje się doświadczenia z istniejących opracowań i sprawdzonych w praktyce standardów tak, że mimo zakładanej uniwersalności można w pewnych koncepcjach odnaleźć kierunki specjalizacji. Znaczna część współczesnych systemów modułarnych jest opracowywana przez producentów systemów mikroprocesorowych, narzucających rozwiązania dostosowane do swoich wyrobów.

Rola organizacji międzynarodowych przy wprowadzaniu i rozpowszechnianiu nowych systemów modułarnych wydaje się obecnie trudniejsza i nieco mniejsza niż w latach poprzednich, a opracowane przez nie propozycje systemów — mniej atrakcyjne niż w swoim czasie np. CAMAC.

Obserwując aktualny stan opracowań proponowanych rozwiązań systemów modułarnych odnosi się wrażenie, że mimo zapewnień o ich uniwersalności i mimo formalnego podobieństwa, są one w pewnym sensie specjalizowane — również ze względu na przewidywane zastosowanie. Przykładowo, systemy VME i AMS [3, 5, 7] są dopasowane w założeniach twórców do wyrobów firmy MOTOROLA (VME) i INTEL (AMS), a system FASTBUS [6] został opracowany dla potrzeb fizyki jądrowej wielkich energii. Systemy P-802 [4] i PROWAY [2] są dostosowane do potrzeb lokalnych sieci komputerowych.

Należy podkreślić, że prace nad nowymi systemami modułarnymi są prowadzone w sposób ciągły, a pierwotnie rozpatrywane propozycje ulegają daleko idącym zmianom, np. do chwili obecnej przedstawiono już cztery wersje systemu AMS i sześć wersji systemu P-896 [1]. Powstał również projekt systemu modułarnego opracowywanego przez firmę INTEL tzw. MULTIBUS-I, a ostatnio pojawiła się jego rozszerzona wersja MULTIBUS-II, co jest szczególnie interesujące ze względu na ogromne rozpowszechnianie układów mikroprocesorowych tej firmy. W niektórych laboratoriach i instytutach Europy Zachodniej wprowadzany jest obecnie system VME (Versamodule Eurocard) opracowany przez zespół firm związanych z koncernem MOTOROLA i oparty o wcześniejsze opracowanie tej firmy tzw. Versabus. Został on zgłoszony, podobnie jak system AMS, do rozpatrzenia jako standard międzynarodowy przez Międzynarodową Komisję Elektrotechniczną (IEC). Jednakże ze względu na silną konkurencję innych rozwiązań modułarnych sprawa wyboru uniwersalnego systemu będzie w dalszym ciągu otwarta.

W kraju obecnie wiele względów przemawia za wyborem systemu VME lub MULTIBUS-II z zastrzeżeniem, że VME jest ukierunkowany głównie na zastosowanie systemu mikroprocesorowego MC 68000. Ostateczna decyzja powinna zostać podjęta w najbliższej przyszłości.

\*

Współczesne systemy modułarne stanowią wewnętrzka-setowe systemy sprzęgające dla zestawów mikrokomputerowych. Przez system sprzęgający rozumie się połączenie między częściami składowymi zestawu mikrokomputerowego (modułami) oraz zasady współpracy zestawów mikrokomputerowych obejmujące:

- protokoły wymiany sygnałów sterujących i danych
- parametry tych sygnałów
- parametry mechaniczne systemu
- zasady zasilania.

## OPIS OGÓLNY

Modułarny system VME jest systemem sprzęgającym mikrokomputer (lub mikrokomputery) i bloki wykonawcze stanowiące niezależne urządzenia przeznaczone do przetwarzania, przechowywania danych pochodzących z urządzeń zewnętrznych i sterowania tymi urządzeniami. W systemie VME zdefiniowano:

- zasady współpracy między dwoma urządzeniami bez zakłócenia indywidualnych działań wewnętrznych każdego z nich
- wymagania dotyczące parametrów elektrycznych i mechanicznych
- protokoły komunikacji między systemem a urządzeniami zewnętrznymi, ze szczegółowym opisem wymaganych funkcji.

Kaseta systemu VME zawiera 20 stanowisk połączonych za pomocą sieci bierniej wykonanej w postaci płyty drukowanej, stanowiącej magistralę systemu. Każde z 20 stanowisk może zawierać więcej niż jedno 96-stykowe złącze trójrzędowe (C96 według normy DIN 41612). Wtyki złącz są montowane do płyty drukowanej magistrali i umieszczone pionowo jedno nad drugim. Gniazda złącz są przymocowane do płytek drukowanych o wymiarach 160×233,68 mm (tzw. podwójna Eurokarta). Płytki z gniazdami stanowią wymienny blok (pakiet) systemu. Długość bloku wynosi 160 mm, a szerokość 20,32 mm. Do przedniej krawędzi płytki mogą być umocowane złącza dodatkowe. Dwa skrajne stanowiska kasety są zajęte przez sieci rezystorowe polaryzujące i dopasowujące linie magistrali.

Modułem w systemie VME nazywa się zespół układów elektronicznych przeznaczony do wykonywania pojedynczej funkcji. Jeden blok (pakiet, płytka drukowana z gniazdami) może zawierać wiele modułów. W dokumencie specyfikacyjnym zdefiniowano następujące rodzaje modułów:

- sterownik (ang. master) — moduł zapoczątkowujący przesyłanie danych, na rys. 1 oznaczony DTB Master (ang. Data Transfer Bus)
- sterownik zgłoszeń (ang. requester) — moduł sterujący, żądający kontroli magistrali
- sterownik przerwań (ang. interrupt handler) — moduł wykrywający przerwania i wywołujący odpowiednie działania
- sterownik podsystemu (ang. master subsystem) — urządzenie zawierające sterownik, sterownik zgłoszeń oraz dodatkowo — blok przerwań lub sterownik przerwań (wykonujący również funkcję bloku przerwań); w skład sterownika podsystemu muszą wchodzić wszystkie pozostałe sterowniki
- blok (ang. slave) — moduł umożliwiający odpowiedź na sygnały przesyłania danych ze sterownika

<sup>1)</sup> NIM — Nuclear Instrumentation Modules

● blok przerwania (ang. interrupter) — moduł generujący sygnał żądania przerwania na żądanie sterownika podsystemu

● blok funkcjonalny (ang. slave subsystem) — urządzenie zawierające blok i blok przerwania, które muszą znajdować się w tym samym pakiecie; bloki przerwania muszą wchodzić w skład bloku funkcjonalnego i sterownika podsystemu (mogą stanowić też niezależne elementy systemu)

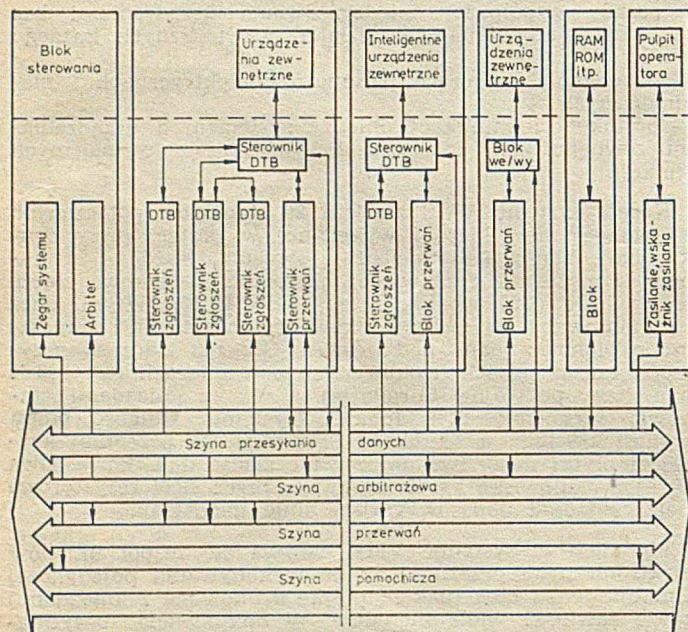
● blok sterowania podsystemu (ang. controller subsystem) — zespół modułów generujących sygnały standardowe i alarmowe magistrali; w każdym systemie musi znajdować się blok sterowania podsystemu, który może zawierać moduły:

- arbitra przydzielającego magistralę
- zegara systemowego
- zerowania systemu
- sygnalizacji zaniku zasilania
- tzw. przeterminowania (ang. time-out).

Każda kaseca systemu VME zawiera blok sterowania podsystemu na stanowisku A1 (lewa strona kasecy). Blok sterowania podsystemu powinien zawierać moduły arbitra i zegara. Pozostałe moduły mogą być umieszczone w dowolnym urządzeniu, np. moduł zerowania systemu może znajdować się przy klawiaturze operatora.

## STRUKTURA MAGISTRALI

W systemie VME można wyróżnić cztery szyny, tj. grupy linii sygnałowych magistrali, łączących poszczególne bloki (mikrokomputery, bloki wykonawcze). Na rys. 1 przedstawiono podstawowe elementy typowego systemu VME i połączenia modułów funkcjonalnych z magistralą. Wewnętrzna konstrukcja bloków jest dowolna. Każdej z czterech szyn magistrali przypisane są określone obszary działania, tzn. zasady komunikowania się bloków między sobą i sposób wykorzystania magistrali: przesyłanie danych, przydział magistrali (arbitraż), określanie priorytetu przerwania, przesyłanie sygnałów pomocniczych.



Rys. 1. Magistrala i podstawowe elementy typowego systemu VME

Do przesyłania danych między blokami systemu służy oddzielna szyna przesyłania danych zawierająca linie danych, linie adresowe oraz linie do przesyłania sygnałów sterujących.

W kasie systemu VME można umieścić kilka sterowników uprawnionych do przesyłania danych. W celu zapewnienia sterowanego dostępu do szyny danych kilku sterownikom, w systemie VME wprowadzono szynę arbitrażu umożliwiającą używanie szyny przesyłania danych w danej chwili tylko przez jeden sterownik.

Określone urządzenia systemu VME mogą zażądać przerwania normalnej pracy magistrali i dokonania obsługi

przez sterownik przerwania. Do tego celu służy szyna przerwania, umożliwiająca operowanie na siedmiu poziomach przerwania.

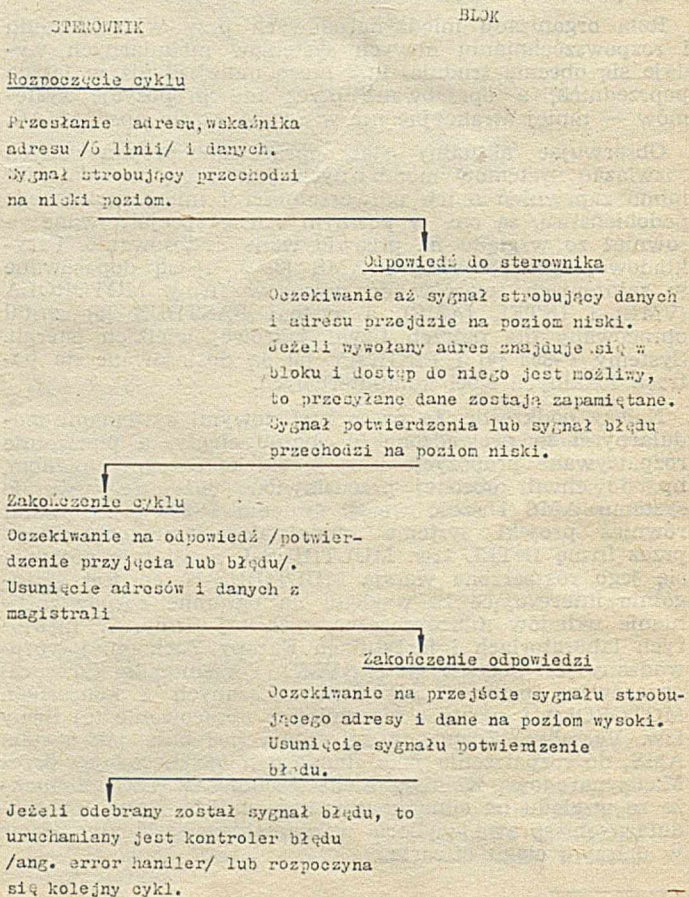
Pozostałe sygnały, jak taktowanie pracy (zegar systemowy), zerowanie początkowe, sygnalizacja zaniku zasilania, sygnalizacja uszkodzeń, są przesyłane po szynie pomocniczej.

Sygnały przesyłane po magistrali mają postać sygnałów napięciowych odpowiadających poziomom napięć TTL (poziom wysoki +2 V, poziom niski 0,8 V) lub narastających i opadających zboczy impulsów przy zmianach poziomu napięcia.

Działanie systemu VME w każdym z wymienionych obszarów (przesyłanie danych, arbitraż, przerwania) jest opisane przez indywidualne protokoły, zawierające również wymagania dotyczące funkcjonowania poszczególnych modułów. Protokoły zawierają ponadto przepisy określające wzajemną współpracę modułów. Moduły komunikują się ze sobą za pomocą zespołów sygnałów (nadawanych i odbieranych) przesłanych po odpowiednich szynach systemu. W protokole opisującym wzajemną komunikację modułów zdefiniowano chwile, w których moduł może sterować i zmieniać poziom napięcia na magistrali oraz chwile i sposób odpowiedzi modułu na odebrane sygnały.

Ogólnie, dla potrzeb analizy pracy systemu VME sygnały podzielono na wewnętrzne i transmitowane. Sygnały wewnętrzne są to zespoły sygnałów wywołujące określone wewnętrzne działania w systemie VME, umożliwiające np. komunikację pomiędzy poszczególnymi modułami, służące do koordynacji jego pracy. Źródłami tych sygnałów są moduły systemu.

Sygnały transmitowane są generowane przez moduły i wysyłane na szynę pomocniczą pod wpływem działania źródeł zewnętrznych. Mogą one być generowane w dowolnej chwili niezależnie od innych działań systemu. W systemie nie zdefiniowano specjalnego protokołu i skutków wywołanych przez te sygnały. Ponieważ sygnały transmitowane nie mają żadnego powiązania z sygnałami magistrali, dla każdego rodzaju sygnału transmitowanego należy poprowadzić indywidualne linie dołączone do wybranych lub wszystkich bloków systemu.



Rys. 2. Kolejność działań przy operacji zapisu danych

## Szyna przesyłania danych

Przesyłanie informacji po magistrali odbywa się asynchronicznie, po wielu liniach równoległych jednocześnie, co pozwala uzyskać znaczną szybkość pracy systemu. Szyna przesyłania danych jest wykorzystywana przez urządzenia sterujące lub układy bezpośredniego dostępu do pamięci (DMA), do przesyłania informacji z lub do pamięci lub urządzeń zewnętrznych. Składa się ona z linii adresowych, linii danych oraz linii sterujących. Liczba linii w każdym z wymienionych zespołów zależy od konfiguracji pracy VME wybranej przez użytkownika. Istnieje, na przykład, możliwość wyboru 16, 24 lub 32 linii adresowych, 8, 16 lub 32 linii danych, linii sygnałów BUS TIME OUT (BTO) i dostępu sekwencyjnego SEQ.

W przewidywanej krajowej konfiguracji systemu proponuje się 16 linii danych, z możliwością adresowania nie mniej niż 24 bitów. Dla bloku sterownika generującego sygnał BUS TIME OUT lub modułu generującego ten sygnał, BTO(n) oznacza, że w przypadku braku odpowiedzi z bloku następuje zaprzestanie cyklu przesyłania danych po n mikrosekundach. Sygnał SEQ jest generowany przez sterownik w celu wymuszania kolejnego przesyłania danych.

W każdej operacji przesyłania danych uczestniczy sterownik i blok. Każda operacja przesyłania jest zapoczątkowana przez sterownik, a adresowany blok musi potwierdzić gotowość do jej wykonania. Po uzyskaniu potwierdzenia przyjęcia przesyłanych informacji sterownik kończy operację. Na rys. 2 przedstawiono diagram ilustrujący kolejność działań przy operacji zapisu danych.

W analogiczny wielostopniowy sposób odbywa się operacja odczytu. Zależności czasowe, kształt sygnałów, kolejność przesyłania, wykorzystanie styków złącz, obciążalność itp. są w systemie VME ściśle zdefiniowane.

## Przydział magistrali — arbitraż

Jednym z zasadniczych zadań systemu VME jest przetwarzanie informacji i związana z tym funkcja przesyłania danych pomiędzy różnymi jego elementami. Ze względu na to, że w kasecie systemu VME znajduje się wiele źródeł sterowania, konieczne było wprowadzenie sterownika przydzielającego szynę przesyłania danych określonym mikrokomputerom zgłaszającym potrzebę wymiany (zapisu lub odczytu) informacji z innym blokiem lub mikrokomputerem. W tym celu wprowadzono szynę arbitrażu składającą się z kilku linii po których przesyłane są odpowiednie sygnały BR (ang. bus request).

System arbitrażu uniemożliwia jednoczesne sterowanie szyną przesyłania danych przez dwa sterowniki, pozwala natomiast zoptymalizować jej wykorzystanie przez wiele sterowników. Zadanie to jest wykonywane przez moduł arbitra umieszczony zwykle w bloku sterowania podsystemu.

System arbitrażu może być zrealizowany w jednym z trzech wariantów:

- arbitraż priorytetowy; pierwszeństwo korzystania z szyny przesyłania danych jest niezmiennie i ustalone przez przypisanie odpowiednich priorytetów liniom sygnałów BR (BR3 — najwyższy priorytet, BR0 — najniższy)
- arbitraż z cyklicznym przeszukiwaniem (ang. round robin select), polegający na cyklicznym sprawdzaniu i kolejnym przydzielaniu dostępu do szyny przesyłania danych zgłaszającym się sterownikom
- arbitraż jednopoziomowy, polegający na przyznaniu bezwzględnego priorytetu jednemu mikrokomputerowi.

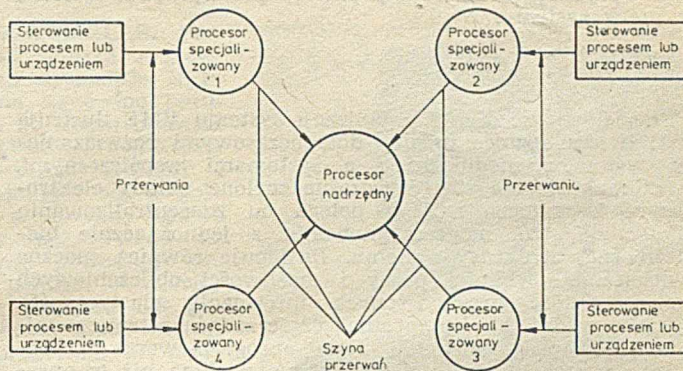
W celu zoptymalizowania kolejności dostępu do szyny przesyłania danych i jej pełnego wykorzystania, w systemie VME przewidziano różne warianty zależności czasowych podczas arbitrażu. Pozwala to na skrócenie czasu trwania poszczególnych operacji i przyspieszenie działania całego systemu.

## Szyna przerwań

W systemie VME przerwania są realizowane w dwojaki sposób:

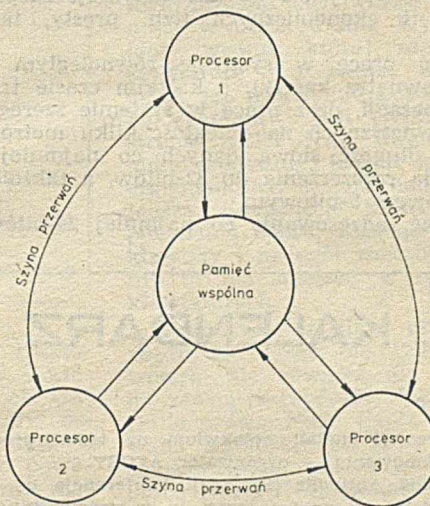
- przez prosty system przerwań, w którym procesor nadrzędny odbiera i obsługuje wszystkie przerwania
- przez zdecentralizowany system przerwań, w którym dwa lub więcej procesorów odbiera i obsługuje przerwania.

Architektura prostego systemu przerwań jest bardziej przejrzysta ze względu na jego podobieństwo do systemów jednoprocessorowych, wyposażonych zwykle w mniej lub bardziej złożone układy przerwań obsługiwane przez odpowiednie podprogramy. W prostym systemie przerwań do ich obsługi stosowany jest jeden procesor kierujący pracą odpowiednich układów sterowanych programowo. Procesor ten może odbierać i obsługiwać przerwania pochodzące z innych podporządkowanych mu procesorów, sterujących pracą urządzeń zewnętrznych lub wewnętrznych (rys. 3).



Rys. 3. Budowa prostego systemu przerwań

Architektura zdecentralizowanego systemu przerwań jest nieco bardziej złożona. Zdecentralizowany system przerwań jest szczególnie wygodny w zastosowaniach, w których część zadań wykonywanych przez jeden procesor może być przekazana do innego, pracującego równolegle. Żaden ze współpracujących procesorów nie jest uprzywilejowany i wszystkie wykonują wspólny program całego systemu, przy czym każdy wykonuje określoną część tego programu i obsługuje tylko te przerwania, które zostały do niego skierowane przez inne procesory systemu. Ponieważ obsługa niektórych przerwań może wymagać wykorzystania całej konfiguracji systemu, procesory mogą komunikować się przez pamięć wspólną, umieszczając w niej podprogramy i informacje o wyniku podjętych działań (rys. 4).



Rys. 4. Budowa zcentralizowanego systemu przerwań

Szyna przerwań składa się z 7 linii sygnałów przerwań, jednej linii łączącej łańcuchowo moduły systemu i jednej linii potwierdzenia przerwania. Szczegółowe zalecenia dotyczące realizacji systemu przerwań w różnych wariantach i konfiguracjach systemu VME powinny być rozpatrywane wspólnie z zaleceniami dotyczącymi arbitrażu.

## Szyna pomocnicza

Szyna pomocnicza służy do przesyłania okresowych sygnałów informacyjnych, diagnostycznych, alarmowych, zezwolenia początkowego itp. W systemie zdefiniowano sygnały:

- zegarowe (SYSCLK)

- brak zasilania (ACFAIL)
- zerowanie systemu (SYSRESET)
- testowanie systemu (SYSFAIL).

Sygnały zerowania i braku zasilania są zwykle powiązane z pulpitem operatora. Sygnały zegarowe są generowane przez niezależny układ w bloku sterowania podsystemu i są wykorzystywane do opóźnienia lub traktowania pracy układów z poszczególnych modułów systemu. Częstotliwość pracy zegara systemu wynosi 16 MHz. Sygnały te nie mają ustalonego związku z innymi sygnałami taktującymi, używanymi w systemie. Sygnały testowania (alarmowe) mogą być wysyłane w dowolnej chwili, po wykryciu usterek w pracy systemu.

\* \* \*

Przedstawione zasady organizacji systemu VME ilustrują podstawowe różnice między dotychczasowymi rozwiązaniami systemów modułowych a systemami współczesnymi, w których stosuje się nowoczesne scalone układy elektroniczne. Zasadnicza zmiana polega na zdecentralizowaniu inteligencji, tzn. na zrezygnowaniu z jednoznacznie hierarchicznej struktury systemu. Następuje również znaczne zwiększenie szybkości pracy i możliwości obliczeniowych systemu (ilość przetworzonych informacji, elastyczność). Ponadto wydaje się, że ogromną część zadań systemu przejmie jego oprogramowanie. Rola oprogramowania w nowych systemach będzie zdecydowanie większa niż w opracowaniach dotychczasowych, wywodzących się z rozwiązań sprzętowych.

Z wymienionych względów, przy wyborze systemu dla potrzeb krajowych należy uwzględnić bardzo wiele czynników, gdyż wybrana koncepcja sterowania zestawów będzie dominować przez szereg lat, a wprowadzenie większych modyfikacji będzie utrudnione. Obecnie, zarysowują się w zasadzie dwa kierunki rozwoju systemów modułowych: prosty, względnie tani i niezbyt rozbudowany system 8-bitowy do różnego rodzaju aparatury eksperymentalnej, oraz system 16-bitowy zbliżony do VME, z możliwością rozbudowy do 32-bitów. Nie przesądza to jednak możliwości budowy mikrokomputerowej aparatury specjalizowanej.

Wybrany system (koncepcja sterowania zestawów) powinien:

- być uniwersalny (zarówno do zastosowań w automatyce przemysłowej jak i do automatyzacji eksperymentów naukowych) i ekonomiczny (tzn. prosty, tani, niezawodny)
- zapewniać pracę w systemie równoległym asynchronicznym (wewnątrz kasyety), o krótkim czasie trwania pojedynczej operacji, oraz pracę w systemie szeregowym dla połączeń zewnętrznych na odległość kilku metrów
- posiadać długość słowa danych co najmniej 16-bitów, z możliwością rozszerzenia do 32-bitów, a także możliwość pracy ze słowem 8-bitowym
- umożliwiać, adresowanie co najmniej 24-bitów

- posiadać standard mechaniczny odpowiadający aktualnym wymaganiom światowym
- posiadać złącza zbliżone do wymagań światowych, np. C96 lub podobne łatwo dostępne (obecnie i w przyszłości) w kraju
- korzystać najwyżej z czterech napięć zasilających, przy napięciu podstawowym +5 V
- zapewniać chłodzenie niewymuszone
- przewidywać pewną liczbę połączeń do dalszego wykorzystania.

Obecnie, podstawowe prace prowadzi się w kierunku adaptacji systemu VME i MULTIBUS-II. Przyjęte dotychczas rozwiązania nie wykluczają możliwości wyboru innego systemu. Jednym z decydujących elementów jest baza elementowa. W systemie VME przewidziano teoretycznie możliwość wykorzystania dowolnego mikrokomputera 16-bitowego, choć był on konstruowany z myślą o zastosowaniu mikroprocesora MC 68000. Należałoby przeprowadzić szczegółową analizę uwzględniającą zarówno rozwiązanie sprzętowe (miejsce na płycie montażowej) jak i programowe, pozwalającą ocenić jakim kosztem (dodatkową konstrukcją, oprogramowaniem) można zastąpić mikroprocesor MC 68000, w szczególności — mikroprocesorem INTEL 8086, który prawdopodobnie będzie łatwiej dostępny.

Dodatkową przeszkodą we wprowadzeniu nowoczesnego systemu modułowego w warunkach krajowych są trudności uzyskania odpowiednich specjalizowanych układów o wielkim stopniu scalenia, oferowanych przez producentów układów (MOTOROLA, INTEL). Układy takie powinny być dostosowane do wymagań proponowanych systemów modułowych, jak np. bezpośrednie dołączenie mikrokomputera do magistrali, rozwiązujące większość problemów związanych z realizacją proponowanych protokołów komunikacyjnych wewnątrz systemu. W ostatnim okresie firma MOTOROLA przedstawiła nową propozycję wprowadzenia dodatkowych szyn do magistrali systemu, znacznie rozszerzających jego możliwości, tzw. VMX do lokalnych połączeń procesora z pamięcią i VMS do transmisji szeregowej. Modyfikacje te bardzo zbliżają system VME do MULTIBUS-II.

#### LITERATURA

- [1] Borrill P.: IEEE P896 — the Futurebus Project. Microprocessors and Microsystems, Vol. 6, No. 9, pp. 489—495 (1982)
- [2] Funk G.: Evaluation of Standard LAN Proposals for Process Control Applications. IEC SC65C WG06, Oct. 1983
- [3] IEC Subcommittee 47B: Proposal of the German National Committee for the AMS Bus System. IEC 47B (Germany) 5 May 1982
- [4] IEEE P802 Local Network Standard. Draft E, 1983
- [5] Jankowski M. T.: KRABUS — sprzęg wewnętrzny dla systemów mikrokomputerowych. INFORMATYKA 1983, nr 1, s. 17—20
- [6] Larsen R. S.: Introduction to the FASTBUS Standard Data Bus. Interfaces in Computing 1982, Vol. 1, No. 1, pp. 19—32
- [7] MOSTEK, MOTOROLA, SYGNETICS-PHILIPS: VME Bus Specification Manual, Oct. 1981.

## KALENDARZ

### Czerwiec

- 4—6, Nicea (Francja): kolokwium na temat predyspozycji w programowaniu — organizator AFCET
- 7—8, Sophia Antipolis (Francja): konferencja na temat jakościowej oceny predyspozycji w programowaniu — organizator: AFCET
- 1—13, Paryż: PROLAMAT — międzynarodowa Konferencja na temat języków programowania dla obrabiarek — organizatorzy: IFIP oraz IFAC
- 17—21, Kopenhaga: XXVI międzynarodowa konferencja TMS — organizator: The Institute of Management Sciences
- 26—29, Rzym: II światowa konferencja na temat międzynarodowego przepływu danych — organizator: Intergovernmental Bureau for Informatics

### Lipiec

- 20—28, Montreal (Kanada): VII międzynarodowa konferencja na temat rozpoznawania obrazów — organizator: International Association for Pattern Recognition

### Sierpień

- 20—28, Montreal (Kanada): VII międzynarodowa konferencja symposium statystyki obliczeniowej — organizator: IASC
- 28—30, Kopenhaga (Dania): EUROMICRO'84 — 10. międzynarodowe symposium na temat mikroinformatyki i mikroprogramowania — organizator: EUROMICRO

### Wrzesień

- 3—17, Londyn: INTERACT'84, międzynarodowa konferencja na temat czynnika ludzkiego w systemach informatycznych oraz współdziałania człowiek-maszyna — organizatorzy: IFIP, IFAC, IFORS oraz IEA
- 10—14, Paryż: VI międzynarodowy kongres cybernetyki i teorii systemów — organizator AFCET oraz World Organization of General Systems and Cybernetics
- 17—21, Paryż: międzynarodowa konferencja „Convention Informatique'84” oraz wystawa „SICOB'84”
- 24—28, Hong Kong: międzynarodowa konferencja pn. „Technologia informatyczna — środek maksymalizacji potencjału gospodarczego krajów azjatyckich” — organizator: SEARCC (South East Asia Regional Computer Confederation)

Artykułem o NEPTUNIE 134 rozpoczynamy prezentację mikrokomputerów rodzimej produkcji. Być może planowany cykl publikacji pozwoli potencjalnym nabywcom na świadomy wybór sprzętu, odpowiadającego zarówno ich potrzebom jak i możliwościom finansowym. Prawdopodobnie ubocznym efektem naszych opisów będzie zwiększenie się długości kolejek oczekujących na sprzęt — ale być może otworzy to oczy niektórym decydom, dla których problem krajowej mikroinformatyki nie istnieje.

NEPTUN zdobył sobie dość szeroką rzeszę entuzjastów wśród naukowców, jako system przydatny w laboratorium. Ewenementem jest przełamanie w tej konstrukcji powszechnego sentymentu do rodziny podzespołów wprowadzonych przez firmę INTEL. Zaoferowało to możliwością wykorzystania oprogramowania, które ze skromnego kontrolera może uczynić NEPTUNA niemal wszechmocnym narzędziem pracy.

## NEPTUN 184

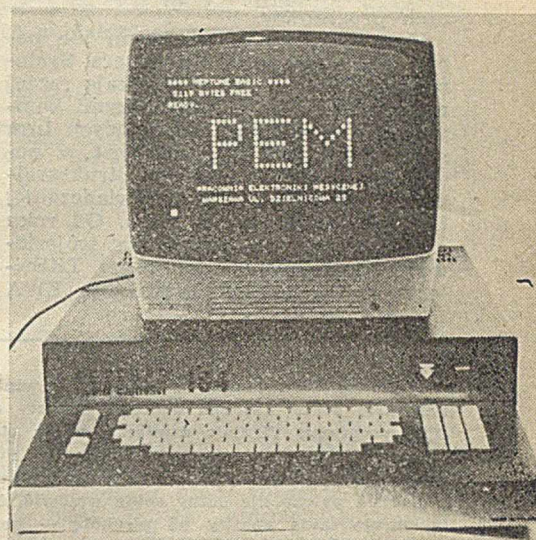
Kontroler systemowy NEPTUN 184 zbudowany został na bazie 8-bitowego mikroprocesora NMOS typu 6502. Przeznaczony jest zasadniczo do sterowania aparatury pomiarowej i kontrolnej. System wyposażony jest w interpreter języka BASIC oraz MACROASSEMBLER dla mikroprocesora 6502. Procedury w języku maszynowym mogą być wywoływane z programów napisanych w języku BASIC lub wykonywane jako niezależne programy.

Użytkownik ma do dyspozycji do 32 KB pamięci RAM, możliwość współpracy z dwoma magnetofonami kasetowymi, sprzęg IEC 625 (inaczej nazywany HPIB), sprzęg do drukarki DZM-180 oraz bogaty zestaw dodatkowych modułów funkcjonalnych (m.in. możliwość dołączenia stacji dysków elastycznych, plottera X-Y oraz czytnika i perforatora taśmy papierowej). Wymienione możliwości sprawiają, że obszar zastosowań kontrolera wykracza daleko poza dziedzinę pomiarów i kontroli. Wykorzystanie w systemie mikroprocesora 6502 pozwala zastosować dla NEPTUNA ogromną ilość oprogramowania napisanego dla mikrokomputerów APPLE i COMMODORE (z nieznacznymi jedynie zmianami).

System operacyjny umieszczony został w pamięci typu ROM i zgłasza się bezpośrednio po włączeniu zasilania. System zgłasza się w programie EDYTOR, który umożliwia wprowadzenie, edycję i uruchomienie programu w języku BASIC. Z poziomu EDYTORA możliwe jest przejście do programu MONITOR, a stąd do

programu MACROASSEMBLER. Program EDYTOR jest jednym z głównych atutów, jakimi dysponuje NEPTUN — pozwala na swobodne poruszanie się kursorem po całym obszarze ekranu. Dzięki swobodzie ruchu kursora, a także funkcjom INSERT i DELETE, tekst znajdujący się w polu ekranu może być w dowolny sposób zmieniany. Zmiany dokonywane na ekranie drogą edycji powodują wyłączenie zmiany w pamięci obrazowej. Dopiero naciśnięcie klawisza CR powoduje, że poprawiona linia zostaje wprowadzona do programu. W przypadku, gdy liczba znaków w linii przekroczy 40, następna linia zostaje logicznie dołączona do poprzedniej, umożliwiając korzystanie z logicznej linii osiemdziesięcioznakowej.

Korzystając z programu plottera ekranowego, można uzyskać na ekranie niemal ciągle obrazy graficzne (rozdzielczość ekranu wynosi 320×200



Klawiatura składa się z 64 klawiszy standardowych oraz 24 klawiszy specjalnych. Każdy ze standardowych klawiszy ma cztery możliwe znaczenia, tak więc bezpośrednio z klawiatury można wygenerować 256 różnych kodów. Podstawowe znaczenie klawiszy oznaczone zostało na klawiaturze. Niestety, specjalne znaczenia nie zostały uwidocznione, co stwarza pewne trudności w użytkowaniu. Wydaje się, że idea, według której zaprojektowano klawiaturę jest bardzo udana, natomiast jej wykonanie pozostawia wiele do życzenia i jest chyba najsłabszą stroną NEPTUNA. Wygląd jej jest mało estetyczny, kształt klawiszy nieergonomiczny, a i mechanika nie najlepsza. Naciśnięcie klawisza wymaga pokonania oporu, różnego dla różnych klawiszy i na ogół znacznie większego niż to ma miejsce w nowoczesnych klawiaturach.

Wbudowany zasilacz dostarcza niezbędnych napięć nie tylko dla kontrolera, ale również dla dołączanych na zewnątrz modułów funkcjonalnych. Do odprowadzenia ciepła nie zastosowano wymuszonego obiegu powietrza, co wprawdzie powoduje bezszmerową pracę NEPTUNA, jednakże przy pracy powyżej kilkunastu godzin zdarzają się kłopoty. Wadę tę można wyeliminować w prosty sposób, realizując wymuszony obieg powietrza w okolicy radiatorów zasilacza (wystarczy najprostszy wentylator).

Bardzo dobrze rozwiązana jest współpraca z magnetofonami. Natomiast współpraca NEPTUNA z modułem dysku elastycznego produkcji MERA-KFAP pozostawia jeszcze dużo do życzenia. Producent obiecuje wprowadzenie nowego systemu dyskowego, umożliwiającego łatwą pracę z dyskami elastycznymi z poziomu języka BASIC. Warto tutaj nadmienić, że oprogramowanie i możliwości NEPTUNA

### Mikrokomputer NEPTUN 184

CPU: 6502  
RAM: 16 KB  
Oprogramowanie: BASIC  
Opcje:  
dodatkowy RAM 16 KB  
dodatkowy EPROM 16 KB lub 32 KB  
program ASSEMBLER 6502  
sprzęg IEC 625 (HPIB)  
sprzęg do drukarki DZM 180  
sprzęg V24  
sprzęg dysków elastycznych  
przetwornik 8×8bit A/C (100 μs)  
Cena: 820 tys. zł.  
Producent: Pracownia Elektroniki Medycznej  
Warszawa-Wawer, ul. Dzielnicowa 25  
tel. 12-07-87  
Czas realizacji zamówienia: ok. pół roku

punktów). Ekran można więc traktować jako zbiór 64 tys. punktów, z których każdy może być w dwóch stanach — białym lub czarnym. Nie ma możliwości wyświetlania punktów o różnych poziomach szarości, co powoduje, że ewentualne przyłączenie monitora kolorowego nie jest rzeczą prostą.

Znaki alfanumeryczne definiowane są w matrycy 8×8 punktów. Na ekranie może być wyświetlone 25 wierszy tekstu po 40 znaków w wierszu.

Pamięć obrazu zajmuje 1 KB. Każda 8-bitowa komórka pamięci obrazowej może zawierać kod jednego z 256 znaków, dostępnych w pamięci RAM generatora znaków. Generator znaków z pamięci stałej ROM jest przepisywany do pamięci RAM w czasie inicjacji systemu. Użytkownik ma możliwość wprowadzenia dowolnego własnego generatora znaków.

są przez producenta praktycznie ciągle ulepszone, przy czym udało się uniknąć takich przeróbek, które wymagałyby wprowadzania poprawek do starych programów.

Uwagi krytyczne w niczym nie pomniejszają wartości Kontrolera Systemowego Neptun 184 i wynikają raczej z porównania go z najnowszymi komputerami osobistymi przodujących firm światowych. NEPTUN 184 jest w posiadaniu Zakładu Badań Strukturalnych Instytutu Fizyki Doświadczalnej UW od ponad półtora roku. Od roku używany jest do sterowania trójosiowego spektrometru neutronów TKS-420 przy kanale nr 9 reaktora EWA

w Świerku. Taki typ pracy wymaga aparatury o długich okresach międzyawaryjnych, gdyż system musi pracować w nieprzerwanych cotygodniowych cyklach 96-godzinnych w ciągu prawie całego roku. W czasie półrocznego okresu eksploatacji NEPTUNA wystąpiła tylko jedna awaria, usunięta przez producenta w ciągu kilku godzin. W przodujących laboratoriach do sterowania spektrometrów podobnej klasy są używane minikomputery PDP-8, wymieniane ostatnio na komputery PDP-11. Zastosowanie Kontrolera Systemowego NEPTUN 184 pozwoliło na zrealizowanie sterowania eksperymentem „on-line” na poziomie

odpowiadającym poziomowi przodujących laboratoriów europejskich.

Wydaje się, że w chwili obecnej NEPTUN jest najatrakcyjniejszym systemem mikroprocesorowym dostępnym na naszym rynku.

**WŁADYSŁAW MINOR**

Wydział Fizyki  
Uniwersytet Warszawski

EO/706/K/84

Artykuł ten został opublikowany w porozumieniu z producentem i przez niego sfinansowany.

W Polsce to legenda, w krajach rozwiniętych zaś — narzędzie pracy każdego dobrego planisty czy zarządcy. Opis programu VisiCalc zamieszczamy z cichą nadzieją, że nie tylko cinkciarze podejmują w Polsce decyzje w oparciu o szczegółową analizę faktów i danych. A już na pewno nie damy sobie wmówić, że ktokolwiek wykonuje takie analizy na piechotę — bez komputera. Rozpowszechnienie oprogramowania mogłoby tu więc uchodzić za papierek lakmusowy pokazujący w jakim stopniu (tak naprawdę, nie w gazetach) realizowana jest u nas racjonalna gospodarka. I tylko bez wykrętów: TABPLAN — czyli polska wersja programu VisiCalc nie podlega reglamentacji i może ją nabyć każda firma za kwotę niewiarygodnie małą w stosunku do korzyści, jakie można uzyskać z racjonalnego gospodarowania.

## VisiCalc

Zapewne wielu Czytelników choć raz w życiu mozolnie przygotowywało tabelkę, którą mógłby równie dobrze drukować komputer (rys. 1). To z pozoru łatwe zadanie komplikuje się, jeżeli zażądamy, aby rubryki miały różną szerokość i wysokość lub — co gorzej — by niektóre elementy tabeli były zależne od innych elementów. Problem rozwiązuje natychmiast VisiCalc, program produkowany (ciągle trudno pojąć, że oprogramowanie jest również dobrą produkcją jak obrabiarki i statki...) przez firmę, która dziś od swego szlagentu nazywa się VisiCorp. A oto podstawy działania i główne możliwości programu, który jest standardowym już wyposażeniem komputerów biurowych.

	Cinkciarz		komputerowy		
Data	27 I 84	3 II 84	10 II 84	17 II 84	Projekcja
Bony (zł)	615.00	640.00	635.00	645.00	655.00
Złoto "2" (tys.zł/g)	5.40	6.00	6.00	6.10	6.20
Złoto PL (%/uncja)	364.10	388.75	391.81	392.17	392.52
Złoto USA (%/uncja)	369.93	387.02	386.25	387.35	388.45

Rys. 1

Spójrzmy na rysunek 2: jest to przedruk z ekranu monitora po uruchomieniu programu VisiCalc. Widzimy przede wszystkim sposób określania wierszy (liczby) oraz kolumn (litery) tabelki. Zwykle za standardową uznaje się macrycę o wymiarach 255×255. Liczba elementów powinna być duża, gdyż niektóre z nich będą pełniły tylko rolę „ozdobników”, umożliwiając — na przykład — zatytułowanie tabeli (na rysunku — pierwszy wiersz), oddzielanie rubryk (wiersze 2 oraz 4) lub optyczne „rozstrzelanie” tabeli (wiersze 6 oraz 9). Abyśmy mogli zdefiniować co chcemy umie-

ścić w odpowiednim segmencie macierzy, program został wyposażony w tzw. okienko, które pozwala podglądać wybrany element (w przykładzie jest to element B11). U góry ekranu widać, że element ten został zdefiniowany jako pewna kombinacja innych elementów macierzy. Oczywiście, może tu również dobrze wystąpić stała liczbową lub znaki alfanumeryczne tworzące komentarz. Przesunięcie okienka poza widoczny na ekranie obszar powoduje automatyczne przesunięcie całej macierzy — w ten sposób możemy oglądać nie tylko wybrany element, ale i jego otoczenie. Tak stało się przy próbie zdefiniowania elementu F8 (rys. 3). Elementy puste nie są wyświetlane.

B11 /F# (V) (4000/3#88/B5#31.1)  
Copyright (c) 1980 Software Arts Inc. V1.20Z

C  
22

	A	B	C	D
1.			Cinkciarz	komputerowy
2.				
3. Data	27 I 84	3 II 84	10 II 84	17 II 84
4.				
5. Bony (zł)	615.00	640.00	635.00	645.00
6.				
7. Złoto "2" (tys.zł/g)	5.40	6.00	6.00	6.10
8.				
9. Złoto PL (%/uncja)	364.10	388.75	391.81	392.17
10.				
11. Złoto USA (%/uncja)	369.93	387.02	386.25	387.35
12.				

Rys. 2

Dochodzimy teraz do sedna całego programu. Otóż wyobraźmy sobie, że chcemy zobaczyć co się stanie, jeżeli jeden element macierzy zostanie zmieniony (rys. 4). Wstawiamy wartość 650 zamiast 645. Kolejny rysunek pokazuje zdawałoby się taką samą tabelkę. Ale podkreślone elementy zmieniły się, choć zmiana danych wejściowych dotyczyła tylko jednej pozycji tabeli (por. rys. 5 i 1). W taki to prosty sposób możemy automatycznie uzyskiwać odpowiedź na pytanie: „Co by było, gdyby...?” Pytanie takie stawiają sobie po wielokroć wszyscy dyrektorzy. Jak widać, komputer ułatwia i przyspiesza znajdowanie odpowiedzi.

By rzeczywiście ułatwiać życie i przyspieszać podejmowanie decyzji VisiCalc posiada wiele opcji. Oto niektóre z nich: możliwość replikacji wzorów określających pewne elementy (w przykładzie — kolumny B, C, D itd. zostały

FB (V) (2#E8-D8)

C  
22

	C	D	E	F
1. Cinkciarz komputerowy				
2.				
3.	3 II 84	10 II 84	17 II 84	Projekcja
4.				
5.	640.00	635.00	650.00	665.00
6.				
7.				
8.	6.00	6.00	6.10	6.20
9.				
10.				
11.	388.75	391.81	389.15	386.49
12.				

Rys. 3



E5 (V) 645  
Value

C  
22

	B.....	C.....	D.....	E.....
1.	Cinkciarz komputerowy			
2.				
3.	27 1 84	3 11 84	10 11 84	17 11 84
4.				
5.	615.00	640.00	635.00	645.00.
6.				
7.				
8.	5.40	6.00	6.00	6.10
9.				
10.				
11.	364.10	388.75	391.81	392.17
12.				

Rys. 4

określone tym samym wzorem, wynikającym z przeliczenia cen). Łatwo możemy wybierać odpowiednie formaty (w przykładzie zastosowano oczywiście format z dwiema cyframi po przecinku, ale gdyby tabela dotyczyła liczby pingwinów na Antarktydzie, to pewnie wolelibyśmy format całkowity). Możemy też zażądać drukowania szerszych lub węższych kolumn. Pozwala to zamieścić mniej lub więcej materiału na danej szerokości papieru. Wreszcie, możemy drukować tylko część tabeli — np. tylko bieżące dane.

Bardziej rozbudowane wersje programu VisiCalc umożliwiają definiowanie całych procedur na zmiennych macierzy, co pozwala np. na dokonywanie regresji liniowej (przewidywanie trendów itp.), a nawet na symulowanie niektórych procesów gospodarczych. Najnowsze wersje programu VisiCalc łączą podstawową ideę tablicy oraz okienka podglądowego z grafiką (wykresy słupkowe danych), połą-

zeniami z innymi komputerami (bazy danych), a nawet z procesorem tekstów. Taki globalny program nazwany LisaCalc (opracowany dla mikrokomputera LISA — patrz INFORMATYKA nr 11, 1983) może współpracować z innymi elementami systemu Lisa (LisaGraph, LisaDraw, LisaList, LisaProject oraz LisaWrite), tworząc najdoskonalsze obecnie biuro w jednej skrzynce!

	Cinkciarz komputerowy				
Data	27 1 84	3 11 84	10 11 84	17 11 84	Projekcja
Bony (zł)	615.00	640.00	635.00	650.00	665.00
Złoto "2" (tys.zł/g)	5.40	6.00	6.00	6.10	6.20
Złoto PL (\$/uncja)	364.10	388.75	391.81	389.15	386.49
Złoto USA (\$/uncja)	369.93	387.02	386.29	387.35	386.45

Rys. 5

W sytuacji, gdy wszystkie stałe są „zienne”, możliwość przewidywania „co by-gdyby” dzięki jednemu przyciśnięciu klawisza jest pokusą nie do odparcia. Nie będziemy wspominać o liczeniu bilansów, list wypłat i innych typowych we wszystkich biurach zajęciach, które spędzają sen z powiek urzędnikom. A może jednak informatyka jest potrzebna???

JAKUB TATARKIEWICZ

## 8255 w praktycznych zastosowaniach

Układ 8255, wprowadzony na rynek przez firmę INTEL, jest jednym z najczęściej wykorzystywanych programowanych układów równoległego wejścia-wyjścia. Projektowany był do wykorzystania w systemach sterowanych przez procesor 8080. Można go jednak z powodzeniem zastosować w konstrukcjach zawierających inne procesory firmy INTEL, a nawet procesory firmy ZILOG — Z80. Sformułowanie „programowany” oznacza, że procesor może zdefiniować różne tryby pracy układu, a także wybrać kierunek transmisji niezależnie dla każdego z trzech portów: A, B i C.

Opis funkcjonalny układu można znaleźć m.in. w [1, 3, 4]. Dlatego też poniżej nie zamieszczamy kolejnego przedruku z katalogu. Skupimy się natomiast na pewnych specyficznych cechach układu, istotnych przy jego stosowaniu. Odpowiednikiem układu 8255 jest produkowany w Polsce MCY 7855.

### Wykorzystanie trybu zerowego

W trybie zerowym każdy z portów A, B i C może być niezależnie zdefiniowany:

- jako wejście służące do badania doprowadzonych sygnałów
- jako wyjście pozwalające wymusić na poszczególnych liniach zadane stany logiczne.

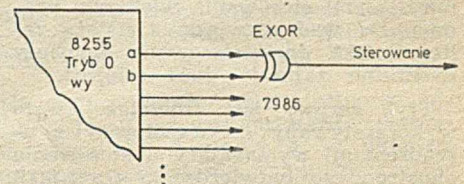
W przypadku portu C można również zdefiniować odmienny kierunek przepływu informacji dla dwóch połówek (po 4 bity) portu: „górną” i „dolną”.

Zdarzają się jednak egzemplarze układu, w których odczytanie informacji, z portu C powoduje zmianę stanów logicznych w połowie zdefiniowanej jako wyjście. Dlatego też należy unikać takiej konfiguracji portu C.

Po wyzerowaniu układu, wszystkie linie portów A, B, C znajdują się w stanie dużej impedancji (tzn. układ nie steruje tych linii). Cechą charakterystyczną układów TTL, nie sterowanych na wejściu, jest interpretowanie takiej sytuacji jako stanu wysokiego (ang. High). Tak więc odbiorniki sterowane przez układ 8255, po podaniu w systemie sygnału zerującego RESET, zachowują się tak, jakby były sterowane stanem H. Po zaprogramowaniu układu w trybie zerowym, na wszystkich liniach zdefiniowanych jako wyjście, pojawia się stan niski (ang. Low). Dopiero zapisanie danej do portu może zmienić stan linii. Stwarza to problemy przy sterowaniu urządzeń.

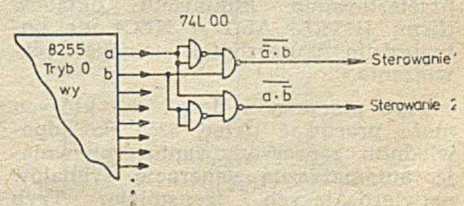
Dla zilustrowania problemu rozważmy następujący przykład. Dwie linie wyjściowe wykorzystywane są do sterowania ważnych zaworów, które w pewnych momentach procesu powinny być otwierane i zamykane. Otwarcie zaworu w złym momencie może spowodować eksplozję. Jeżeli stan H na wyjściu oznacza zawór zamknięty (a więc wyzerowanie systemu mikroprocesorowego jest niegroźne, bo oba zawory są zamknięte), to zaprogramowanie układu w nieodpowiednim momencie może spowodować, że „wyleci-

my w powietrze” zanim system zdąży zapisać bajt danych zamykający zawory. Dla uniknięcia takiej sytuacji stosuje się specjalne rozwiązania sprzętowe.



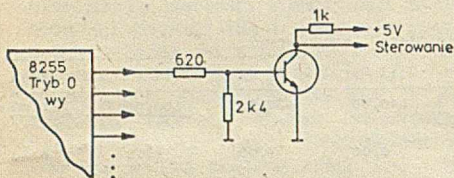
Rys. 1. Na wyjściu sterującym pojawia się stan logiczny 1 tylko gdy  $ab \vee ab$

Najbardziej oczywistą metodą jest zastosowanie do sterowania urządzenia zewnętrznego dwóch linii wyjściowych. Jeżeli na linii podane są różne stany logiczne, to urządzenie jest uruchamiane (np. otwiera się zawór). Na rysunku 1 przedstawiono rozwiązanie wykorzystujące bramkę typu EXOR (ang. EXCLUSIVE OR). Stanem aktywnym wyjściowej linii sterującej jest stan H. Rozbudowując nieco układ można za pomocą dwóch linii sterować dwoma urządzeniami (rysunek 2). Wadą takiego rozwiązania jest brak możliwości załączania obu urzą-



Rys. 2. Stan aktywny na wyjściu 1 pojawia się dla  $ab = 01$ , na wyjściu 2 dla  $ab = 10$

dzeń równocześnie. Ze względu na niewielką obciążalność wyjść układu 8255, zastosowane bramki powinny być typu 74LS00 lub 74L00.



Rys. 3. Stan aktywny na wyjściu pojawia się tylko gdy wyjście portu jest w stanie logicznym 1

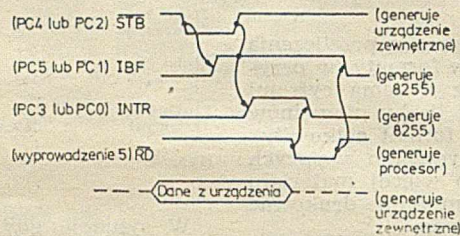
Trzecie rozwiązanie (rysunek 3) pozwala wykorzystać każdą linię układu 8255 niezależnie od pozostałych. Jeżeli linia sterująca znajduje się w stanie dużej impedancji, tranzystor jest zatkany i wyjściowa linia sterująca znajduje się w stanie H. Odpowiednio dobrana polaryzacja tranzystora zapewnia, że gdy linia wyjściowa układu 8255 znajduje się w stanie L, tranzystor nadal pozostaje zatkany. Przejście linii sterującej w stan H powoduje odetkanie tranzystora, który wchodzi w obszar pracy w nasyceniu. W związku z tym wyjściowa linia sterująca znajduje się w stanie L. Zastosowanie tranzystora ułatwia ponadto współpracę z odbiornikami wymagającymi sterowania większym prądem. Do wad rozwiązania należy zaliczyć fakt, że zastosowane elementy dyskretne zajmują dużo miejsca.

Tryb zerowy jest również często wykorzystywany do współpracy z urządzeniami wymagającymi stosowania obustronnego potwierdzenia transmisji (ang. hand shake). Obsługa linii kontrolno-sterujących odbywa się wtedy programowo, co upraszcza w znacznym stopniu dopasowanie się do różnych urządzeń.

Dla uniknięcia błędnych stanów należy wykorzystać jedno z przedstawionych wcześniej rozwiązań. Uwaga: w [1] podano metodę programowego rozwiązania problemu związanego z przejściem wyjść w stan L bezpośrednio po zaprogramowaniu układu 8255. Polegać ma ona na zapisie odpowiedniej danej do portu przed zaprogramowaniem układu 8255. Niezbędny jest jednak wybór konfiguracji, w której port C zdefiniowany jest jako wejście. Niestety, przetestowanie kilkunastu układów (firm: INTEL, AMD i NEC) nie potwierdziło skuteczności takiego rozwiązania.

Tryb pierwszy i drugi, w których może pracować układ 8255 po odpowiednim zaprogramowaniu, zapewniają automatyczną generację sygnałów potwierdzających transmisję. Tryb pierwszy wykorzystywany jest albo do transmisji wyjściowej, albo do wejściowej, natomiast w trybie drugim port A w zależności od sygnałów na

liniach kontrolnych służy jako wejście lub wyjście. Ze względu na skomplikowane sterowanie, tryb drugi wykorzystywany jest stosunkowo rzadko. Dlatego też poniżej opiszemy jedynie zagadnienie związane z wykorzystaniem trybu pierwszego. Zasygnalizowane problemy występują również w bardzo zbliżonej formie dla trybu drugiego.



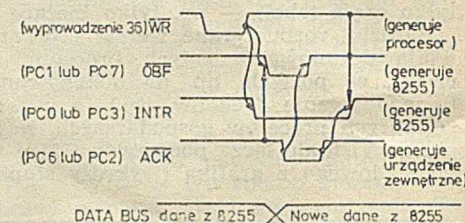
Rys. 4. Wprowadzanie danych z automatycznym generowaniem sygnałów sterujących. Sygnał  $\overline{STB}$  (ang. Strobe) informuje o obecności danych. Sygnał  $\overline{IBF}$  (ang. Input Buffer Full) spełnia rolę potwierdzenia.  $\overline{INTR}$  (ang. Interrupt) — sygnał zgłoszenia przerwania do systemu.  $\overline{RD}$  (ang. Read) — sygnał odczytu danych przez procesor

Na rysunku 4 przedstawiono zależności czasowe przy przekazywaniu informacji przez urządzenie zewnętrzne do systemu. Obecność danych na magistrali komunikacyjnej powinna zostać zasygnalizowana przez stan L na wejściu  $\overline{STB}$ . W odpowiedzi układ 8255 automatycznie wymusza stan H na wyjściu  $\overline{IBF}$ . Jest to sygnalizacja dla urządzenia zewnętrznego, że może zakończyć cykl, przywracając stan H na linii  $\overline{STB}$ . Według wymagań katalogowych, urządzenie zewnętrzne powinno jeszcze przez co najmniej 180 ns sterować linie danych po przywróceniu stanu H na linii  $\overline{STB}$ . Jak wskazuje praktyka, czas ten jest podany z dużym zapasem, gdyż poprawne informacje wprowadzane są przez układ, nawet gdy linie danych przestają być sterowane już po ok. 20 ns. Odczyt wprowadzonych danych może nastąpić w dowolnym momencie, gdyż są one pamiętane w wewnętrznym rejestrze układu. Po odczycie, na wyjściu  $\overline{IBF}$  automatycznie wymuszony jest stan L, co sygnalizuje, że urządzenie współpracujące może przejść do realizacji następnego cyklu transmisji.

Stan linii  $\overline{IBF}$  może być odczytywany przez procesor w rejestrze wskaźników. Warto jednak ograniczyć korzystanie z tego wskaźnika tylko do specyficznych sytuacji. Nazwa zastosowana przez producenta jest myląca, gdyż wprowadzona do układu informacja nie musi jeszcze odpowiadać rzeczywistej danej. Urządzenie współpracujące spełni wymagania [3], gdy przekaże informację na szynę na moment przed przejściem  $\overline{STB}$  w stan H. Poza tym, nawet gdy w chwili przejścia  $\overline{IBF}$  w stan H do portu wprowadzane są już prawidłowe dane,

zrealizowanie odczytu przez procesor może spowodować przejście linii  $\overline{IBF}$  w stan L, zanim urządzenie współpracujące wymusiło ponownie stan H na linii  $\overline{STB}$ . Może to spowodować zawieszenie dalszej transmisji.

Wymienione niedogodności wynikające z wykorzystania  $\overline{IBF}$  można wyeliminować, odczytując wskaźnik związany z linią  $\overline{INTR}$ . Aby było to możliwe, trzeba przed rozpoczęciem transmisji zapisać do rejestru kontrolnego układu 8255 odpowiedni kod odblokowujący wewnętrzny przerzutnik  $\overline{INTE}$  (ang. Interrupt Enable). Należy podkreślić, że „normalny” zapis do portu C nie wpływa na stan tego przerzutnika, mimo że teoretycznie zapis do rejestru kontrolnego i do portu C oddziałują na ten sam bit.



Rys. 5. Wysyłanie danych z automatycznym generowaniem sygnałów sterujących.  $\overline{WR}$  (ang. Write) sygnalizuje zapis danych przez procesor.  $\overline{OBF}$  (ang. Output Buffer Full) sygnalizuje, że dane na szynie mogą być odczytywane przez urządzenie zewnętrzne.  $\overline{ACK}$  (ang. Acknowledge) potwierdza wprowadzanie danych przez urządzenie zewnętrzne.  $\overline{INTR}$  (ang. Interrupt) — sygnał zgłoszenia przerwania do systemu

Na rysunku 5 przedstawiono zależności czasowe przy przekazywaniu informacji z systemu do urządzenia zewnętrznego. Po zapisaniu danych przez procesor przekazywane są one na magistralę komunikacyjną i automatycznie wymuszony jest stan L na linii  $\overline{OBF}$  (nieco wcześniej linia zgłoszenia przerwania  $\overline{INTR}$  przechodzi w stan pasywny L). W odpowiedzi urządzenie zewnętrzne wymusza stan L na linii  $\overline{ACK}$  i rozpoczyna wprowadzanie danej. Układ 8255 automatycznie przywraca stan H na linii  $\overline{OBF}$ . Urządzenie zewnętrzne kończy wprowadzanie danej i sygnalizuje to ponownym wymuszeniem stanu H na linii  $\overline{ACK}$ . Układ 8255 może teraz przyjąć następną daną co sygnalizowane jest przez stan aktywny H na linii  $\overline{INTR}$ .

Stan linii  $\overline{OBF}$  może być odczytywany przez procesor w rejestrze wskaźników. Jednak, podobnie jak przy wprowadzaniu informacji nie wskazane jest wykorzystywanie  $\overline{IBF}$ , tak tutaj wykorzystanie  $\overline{OBF}$  może być powodem błędnego działania. Zapisanie nowej danej przez procesor, gdy  $\overline{OBF}$  przejdzie w stan H, może spowodować zmianę informacji podanej na magistralę komunikacyjną przed zakończeniem wprowadzania przez urządzenie zewnętrzne. Ana-

logicznie jak przy wprowadzaniu danych, lepiej jest wykorzystać wskaźnik odpowiadający linii INTR, po uprzednim odblokowaniu przerzutnika INTE.

Wykorzystanie trybów pracy układu 8255, w których sygnały sterujące generowane są automatycznie, wymaga z reguły zastosowania bardziej lub mniej skomplikowanych układów dopasowujących. Czasami spróbowadza się to jedynie do „odwrócenia” sygnałów na któreś z linii, często jednak wymaga zastosowania układów zmieniających relacje czasowe.

Praktycznie, tryb pierwszy i drugi wykorzystuje się w systemach, w których transmisja obsługiwana jest przez przerwania (w trybie zerowym układ 8255 nie generuje sygnałów przerwań). Przy programowym przeglądaniu wskaźników, rozwiązania „automatyczne” nie przyspieszają

przebiegu transmisji, a program obsługi jest tylko nieznacznie krótszy.

Tryb drugi stosowany jest w przypadkach gdy magistrala wykorzystywana jest do transmisji „w obie strony”, a użycie oddzielnych portów jest niemożliwe. Wykorzystanie trybu zerowego w tym przypadku wymagałoby przeprogramowywania układu przy zmianie kierunku transmisji.

AJP

## LITERATURA

- [1] An Introduction To Microcomputers. Vol 3, OSBORNE/Mc Graw-Hill
- [2] Badźmirowski K, Pleńkos J., Piestrzyński W.: Systemy Mikroprocesorowe. WNT, 1981, str. 230-234
- [3] Intel Component Data Catalog 1980. AFN-01300A-1, 3065 Bowers Avenue, Santa Clara, CA 95051
- [4] Programowany Równoległy Układ Wejścia/Wyjścia MCY 7855. Instytut Technologii Elektronowej, Al. Lotników 32/46, Warszawa.

- Piotr Lindner jest posiadaczem mikrokomputera SHARP PC 1500. Prowadzi biuro usług matematycznych AKSJOMAT. Niezależnie od tego, zainspirowany przykładem warszawskiego ABAKUSA, próbuje założyć w Łodzi klub mikrokomputerowy. Jeżeli uda się zdobyć odpowiedni lokal (rozmowy w toku), klub powinien rozpocząć działalność od września. Podstawowym sprzętem będzie na razie mikrokomputer ZX81. Poszukiwani są chętni do współpracy — w szczególności osoby dobrze znające ZX81, programowany w języku ASSEMBLER, Z80. Kontakt: 91-320 Łódź, ul. Zgierska 142 m. 208; tel. 55-14-45.
- Otrzymujemy wiele telefonów od użytkowników ZX81, którzy poszukują oprogramowania użytkowego (np. ASSEMBLER, FORTH). Prosimy więc osoby dysponujące tego typu oprogramowaniem o udostępnienie go. Oferty chętnie opublikujemy.
- Bieżące informacje prosimy przysyłać — jak najszybciej — pod adresem INFORMATYKI: 00-041 Warszawa, ul. Jasna 14/16 pok. 244; tel. 27-71-40.

W pierwszym wydaniu mikroKLANU zamieściliśmy rozwiązanie umożliwiające wykorzystanie w systemie mikroprocesorowym pamięci dynamicznych. Stosunkowo skomplikowany układ wymagał pewnej wprawy od osób, które chciałyby go zrealizować. W tej sytuacji wielu początkujących konstruktorów decyduje się na stosowanie statycznej pamięci RAM, co niejako automatycznie zmusza do ograniczenia wielkości pamięci o swobodnym dostępie (względnie ekonomiczne). Istnieje jednak rozwiązanie pośrednie, czyli tzw. pamięci pseudostatyczne. Eliminują one konieczność multipleksowania adresów, stosowania układów generujących RAS i CAS, a także problem rozstrzygnięcia konfliktu między odświeżaniem a dostępem do pamięci przez procesor.

Opisane poniżej układy trzeba, niestety, sprowadzać za twarde waluty, ale — jeśli wierzyć producentom — są warte żądanej za nie kwoty.

## Pamięci pseudostatyczne

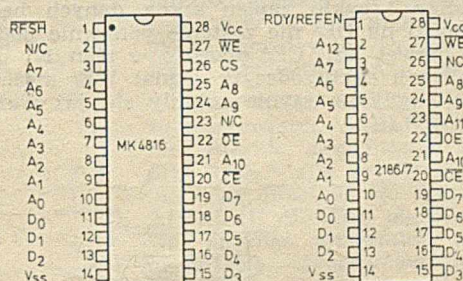
Firma MOSTEK, jako jedna z pierwszych, zaproponowała pamięć dynamiczną zintegrowaną z kontrolerem. Układ MK4816 umieszczony jest w 28-wyprowadzeniowych obudowach, co pozwala na wykorzystanie zamiennie z pamięciami ROM i PROM. Jak przystało na nowoczesną konstrukcję, wymaga on pojedynczego napięcia zasilającego (+5 V ± 10%). Układ wyposażony jest w matrycę pamięci o organizacji 2Kx8bitów. Do sterowania pracą układu wykorzystuje się pięć sygnałów: CE/ (ang. Chip Enable), CS (ang. Chip Select), OE/ (ang. Output Enable), WE/ (ang. Write Enable) i RFSH/ (ang. Refresh). Stan jedenastu linii adresowych jest zapamiętywany w układzie z chwilą przejścia w stan 0 sygnału CE/. W tym momencie linia wybierająca układ CS powinna być w stanie 1. Jeśli linia zapisu WE/ pozostanie nieaktywna (stan logiczny 1) zrealizowany zostanie cykl odczytu. Aby dane zostały wpisane na szynę systemową, musi zostać podany sygnał OE/ (stan logiczny 0). Zapis danej do pamięci odbywa się przez wymuszenie stanu logicznego 0 na wejściu WE/.

Odświeżanie zawartości pamięci może być realizowane na kilka sposobów. Każdy zapis lub odczyt powoduje automatycznie odświeżanie jednego z rzędów matrycy pamięci

(określonego przez sygnały na liniach adresowych A<sub>0</sub>...A<sub>6</sub>). Sposób ten może być wykorzystany w niektórych rozwiązaniach, gdzie gwarantowane jest odwołanie do każdego z rzędów nie rzadziej niż co 2 ms (np. pamięć obrazowa). W typowych zastosowaniach wygodniej jest wykorzystać metodę automatycznego odświeżania się kolejnych rzędów matrycy, przy każdorazowym podaniu aktywnego impulsu na wejście RFSH/. Jeśli jako procesor wykorzystywany jest układ Z80, to stosowanie pamięci 4816 niczym nie różni się od stosowania pamięci statycznych (oczywiście należy połączyć wyjście RFSH/ procesora z wejściem RFSH/ pamięci).

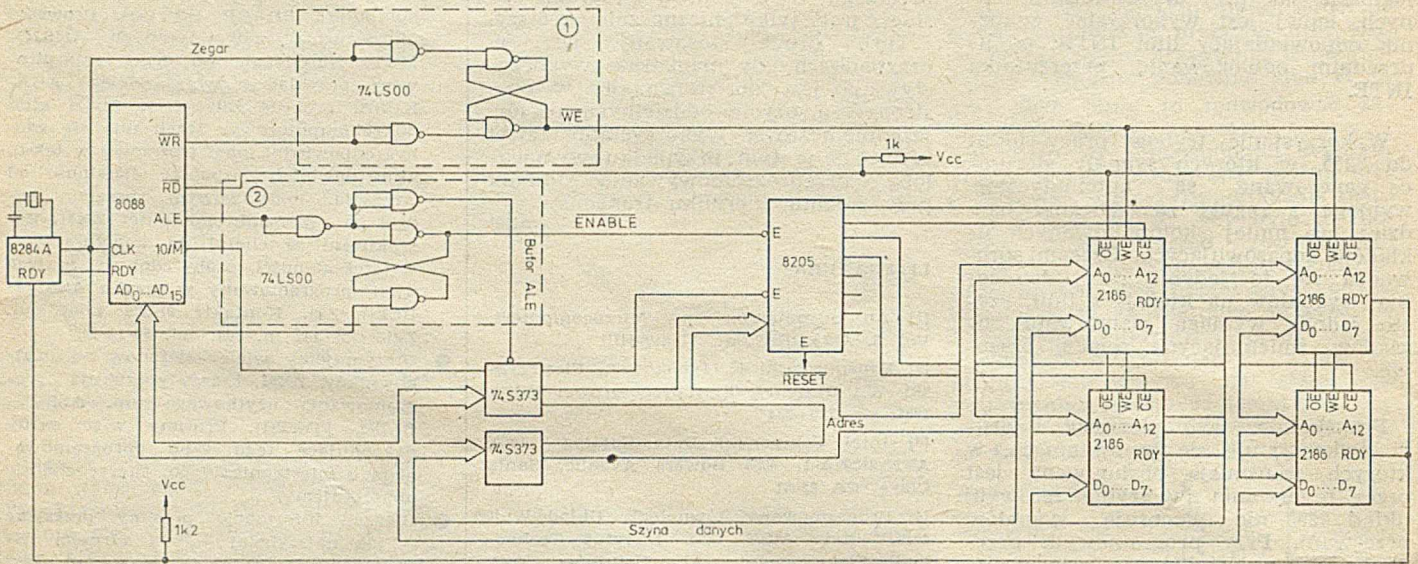
Szczególnie wygodną cechą pamięci jest możliwość samoczynnego odświeżania. Jeżeli na wejściu RFSH/ wymuszony jest przez ok. 20 μs stan logiczny 0, w układzie inicjowane jest samoczynne odświeżanie kolejnych rzędów (mniej więcej co 15 μs kolejny rząd). Cecha ta może być wykorzystana w systemach, gdzie stosowane jest sprzętowe rozwiązanie pracy krokowej, lub też gdy z powodu awarii zasilania niezbędne jest podtrzymanie informacji w pamięci przez zasilanie baterijne (układ 4816 pobiera tylko 5 mA prądu zasilającego przy pracy w trybie „standby”).

Ostatnio firma INTEL wprowadziła na rynek nowy typ pamięci pseudostatycznych. Nazwany on został iRAM (od ang. integrated Random Access Memory) i wyposażony w matrycę pamięci o organizacji 8Kx8 bitów. Podobnie jak omawiany układ firmy MOSTEK, pamięć firmy INTEL produkowana jest w obudowach z 28 wyprowadzeniami. Na rysunku 1 zestawiono rozkład wyprowadzeń dla obydwu układów. Sygnały sterujące pracą układu 2186/7 to: CE/



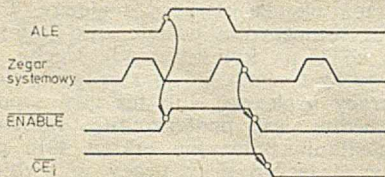
Rys. 1. Topologia wyprowadzeń dla układów MK4816 (MOSTEK) i 2186/7 (INTEL). (Uwaga, powinien być na rysunku RDY/REFEN. Przepraszamy).

1) CE/ — zapis ten jest równoważny z zapisem  $\overline{CE}$ ; w całym artykule (poza rysunkami) przyjęto taką notację



Rys. 2. Przykładowe zastosowanie układów 2186 w systemie wykorzystującym mikroprocesor 8088; cztery układy pamięci dają 32 KB RAM

OE/, WE/ i REFEN/ (ang. Refresh Enable). Podobnie jak w przypadku 4816, podanie stanu logicznego 0 na wejście CE inicjuje cykl wykonywania operacji. Bardzo ważne jest, aby sygnał podawany na wejście CE/ był wolny od fałszywych impulsów (ang. glitch) spowodowanych np. przełączaniem dekodera lub buforów, gdyż może to wywołać niekontrolowany cykl operacji. Przykładowa metoda wytworzenia „gładkiego” sygnału CE/ przedstawiona jest na rysunku 2 (blok 2). Sygnał ENABLE/ generowany jest dopiero wówczas, gdy poprawny adres został wpisany w rejestr buforujący. W ten sposób aktywny sygnał na wyjściu dekodera 8205 pojawia się dopiero w chwili, gdy wszystkie linie przyjęły żądany stan. Przebiegi czasowe układu przedstawiono na rysunku 3.



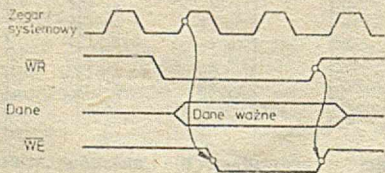
Rys. 3. Przebiegi czasowe dla bloku 2 z rysunku 2

Po rozpoczęciu cyklu przez podanie stanu logicznego 0 na wejście CE/ można zrealizować:

- cykl odczytu (na wejściu OE/ należy wymusić logiczne 0)
- cykl zapisu (na wejściu WE/ należy wymusić logiczne 0)
- cykl „fałszywy” (ang. FMC — False Memory Cycle).

Przy realizacji cyklu zapisu szyna danych będzie sterowana, dopóki na OE/ nie zostanie ponownie podany stan logiczny 1 (nawet jeśli CE/ przeszło w stan 1).

Przy realizacji cyklu odczytu sygnał WE/ może być podany dopiero, gdy na szynie ustaliły się już ważne dane.



Rys. 4. Przebiegi czasowe dla bloku 1 z rysunku 1

W wielu przypadkach konieczne jest więc opóźnienie generowanego przez procesor strobu WR/, zanim zostanie on podany na wejście WE/. Przykładowe rozwiązanie tego problemu pokazano na rysunku 2 (blok 1). Przebiegi czasowe w układzie ilustruje rysunek 4.

Cykl fałszywy jest realizowany wtedy, gdy na żadne z wejść OE/ lub WE/ nie podany zostanie stan logiczny 0. Jeżeli cykl fałszywy jest wykorzystywany dla odświeżenia pamięci, adresy na liniach wejściowych układu powinny być ustalone przed wymuszeniem stanu logicznego 0 na wejściu CE/.

Ponieważ w przypadku pamięci 2186 odświeżanie realizowane jest samoczynnie przez układ, może się zdarzyć, że CE/ zostanie wysterowane w trakcie wewnętrznego cyklu odświeżania. W takiej sytuacji na linii RDY (nóżka 1 układu) pojawia się stan logiczny 0, który może zostać wykorzystany do chwilowego wstrzymania pracy mikroprocesora. W momencie gdy zakończony zostanie wewnętrzny cykl odświeżający, linia RDY ponownie przejdzie w stan 1, co zezwoli mikroprocesorowi na dokończenie operacji. Na rysunku 2 przedstawiono przykładowy sposób realizacji współpracy pamięci 2186 z systemem wykorzystującym mikroprocesor 8088. Wyjścia RDY poszczególnych układów pamięci są sumowane galwanicznie (ang. Wired OR). Układ 8088 pracuje w tzw. minimalnej konfiguracji (jako system jednoprocessorowy). Przy zastosowaniu wersji 5 MHz system może pracować z pełną prędkością, tzn. bez generowania cykli opóźniających przy każdym dostępie do pamięci.

Firma INTEL oferuje też siostrzaną wersję układu pamięci pseudostaticznej o numerze 2187. Różni się ona tym, że na pierwszej nóżce zamiast wyjścia sygnału RDY jest wejście sygnału REFEN/. Przy każdorazowym wymuszeniu stanu logicznego 0, na wejściu REFEN/ inicjowany jest cykl odświeżający, przy czym wewnętrzny licznik generuje adresy kolejnych rzędów. Układ ten jest szczególnie użyteczny przy współpracy z mikroprocesorami generującymi sygnał odświeżania (np. Z80) lub gdy w systemie nie ma możliwości wstrzymywania pracy mikroprocesora.

AJP

## LITERATURA

- [1] The Designers Guide to iRAMs. Solutions, A Publication of Intel Corporation, March/April 1983
- [2] Fallin J. J., Righter W. H.: Designing Memory Systems with the 8Kx8 iRAM. Nota aplikacyjna firmy INTEL AP-132
- [3] Memory Data Book and Designers Guide. MOSTEK, 1980.

Informacja o nowym „dziecku” Sinclair’a obiegła warszawskie lobby mikroinformatyczne lotem błyskawicy. Wiadomość o 32 bitach działała niczym Pershing. Ale nie tylko w 32-bitowej arytmetyce rzecz. Wszystko wskazuje na to, że mamy do czynienia z mikrokomputerem o możliwościach porównywalnych z często u nas spotykanym SM-4. Tyle że za cenę bliską połowy miesięcznej pensji Zachodnioeuropejczyka. Niezawodności, a tym bardziej rozmiarów — nie ma nawet co porównywać. Jest jednak coś jeszcze bardziej uderzającego: w roku rozruchu miesięcznika produkcja wyniesie... dwadzieścia tysięcy egzemplarzy. A nie jest to konstrukcja różniąca się od ZX SPECTRUM wyłącznie kształtem obudowy.

Zgoda, trudno porównywać się nam z firmami amerykańskimi — wiadomo, mocarstwo. Lecz tu mamy do czynienia z krajem, o którego trudnościach gospodarczych codziennie donosi prasa PRL. Genialny Mr Sinclair? Być może też.

I jeszcze jedna ciekawostka. Spodziewając się olbrzymiego popytu na całym świecie, Sinclair rozważa możliwości produkowania QL w różnych punktach naszego globu. Mówi się nawet o ... Chinach. Lecz — mimo że składamy ZX81 — nie mówi się o Polsce. Zostawmy to bez komentarza.

Opisawszy poniżej zasadnicze cechy mikrokomputera SINCLAIR QL, nie uznaliśmy sprawy za załatwioną. W miarę napływu bardziej szczegółowych informacji, będziemy się nimi dzielić z Czytelnikami.

## Sinclair QL

London, druga połowa stycznia 1984. Sir Clive Sinclair prezentuje swoje nowe dzieło — tym razem nie jest to zabawka dla dzieci czy hobbystów, ale komputer profesjonalny. Świadczą o tym nazwa QL — Quantum Leap — czyli przeskok o kilka poziomów możliwości użytkowych.

QL skonstruowano z mikroprocesora MOTOROLA 68008, architektura wewnętrzna 32-bitowa, szyna danych 8-bitowa) uzupełnionego układem INTEL 8049 do sterowania klawiaturą. Cztery inne układy scalone (PLESSEY, FERRANTI, NCR i SYNERTEK) realizują funkcje sterowania wyświetlaniem na ekranie, pamięcią i mikrokasetami.

Wymiary: 138 × 46 × 472 mm, waga: 1,38 kg — czyli zachowano formę „klawiaty z rozumem”.

Pamięć RAM: 128 K bajtów, rozszerzalna do 640 KB (32 KB zarezerwowane na pamięć ekranu). Pamięć ROM: 32 KB (rozszerzalna do 64 KB) zawiera SUPER-BASIC i system op-

racyjny Q-DOS (opracowanie własne Sinclaira; system wieloprogramowy, ale nie wielodostępny).

Rozdzielczość ekranu (monitor albo odbiornik telewizyjny): 512 × 256 punktów (przy czterech kolorach) albo 256 × 256 (przy ośmiu). Można wyświetlać po 85 znaków alfanumerycznych w 25 liniach. Klawiatura QWERTY 65-klawiszowa. Dwa „microdrive-y” (jednostki mikrokaset) o pojemności 100 KB każdy.

Cena zestawu podstawowego: 399 funtów szterlingów (1 funt — 1,4 dol. USA).

Oprócz tego dziewięć podłączeń dla dodatkowych układów — np. sześciu mikrokaset, płytki ROM, dwóch sprzęgów RS 232C, 1 M bajta RAM, sieci lokalnej (mogącej połączyć 64 mikrokomputery QL lub SPECTRUM) i jednego lub dwóch drążków sterujących.

Sir Clive nie byłby sobą, gdyby jego konstrukcja nie odróżniała się od

innych. Na przykład QL nie ma tak modnej ostatnio myszki ani klawisza kasowania, a klawiatura liczy tylko pięć klawiszy funkcyjnych. W istocie Sinclair starał się zredukować ilość opcji i operacji, by uczynić QL dostępnym dla szerokich rzesz... nieprogramistów.

Podstawowym językiem programowania pozostał BASIC, zwany teraz SUPER-BASICIEM. Poszerzono nieco jego możliwości, m.in. o wykorzystanie okienek, tzn. jednoczesne wyświetlanie informacji z różnych programów (a więc jednak coś z najnowszej mody). Oprócz tego, użytkownik dostaje cztery gotowe programy (wliczone w cenę podstawową): Q Quill (przetwarzanie tekstów), QL Abacus (modelowanie ekonomiczne), QL Archive (zarządzanie bazą danych) i Q Easel (grafika). Wszystkie mają znormalizowane menu — zwłaszcza Q Quill — do poziomu debiutantów. Natomiast system operacyjny pozwala na jednoczesne wykonywanie dwudziestu programów, z których cztery mogą współdziałać ze sobą.

Jak by tego było za mało, Sinclair przyznał się, że pracuje już nad rozszerzeniem pamięci 0,5 MB, kompilatorem PASCALA, ASSEMBLEREM, PROLOGIEM, emulatorem terminala, sprzęgami: analogowo-cyfrowym, dysku Winchester (!), modemu i drukarki, a także — nad generatorem dźwięków.

Producent zakłada wytwarzanie tego modelu w liczbie 20 tys. szt. miesięcznie, z perspektywą osiągnięcia w przyszłym roku nawet 100 tys. szt.

Jednocześnie tworzy się klub, a raczej QLUB (QL User Bureau). Składka roczna 35 funtów daje prawo do regularnego biuletynu informacyjnego i otrzymywania aktualnych wersji podstawowych programów.

Oprac.

MAREK SOBCZYK

na podst. 01 Informatique hebdo  
z dn. 23.01.1984

APPLE II czy nawet III — to już przeszłość; a jednak ciągle sprzedawane są w dużych ilościach. Dlaczego? Przyczyną jest olbrzymia ilość oprogramowania (krążącego również po Polsce), które pozwala nie rozwiązywać każdego problemu od początku. Ten fakt powinni wziąć pod uwagę wszyscy noszący się z zamiarem sprowadzenia sobie mikrokomputera — tym bardziej, że dzięki szerokiej gamie rozszerzeń sprzętowych APPLE jest w stanie zrealizować praktycznie wszystkie zadania, jakie mogą być przydatne dla przeciętnego użytkownika.

Warto też wspomnieć, że mikrokomputer bardzo zbliżony do APPLE II... produkowany jest w Polsce (czegoż nie potrafi prywatna inicjatywa!). Niestety, liczba zamówień przekroczyła możliwości produkcyjne na kilka najbliższych lat. Warsztat rzemieślniczy, nawet najlepiej zorganizowany, nie zastąpi przecież fabryki.



O jabłku opowieść (1)

Kalifornia w 1976 roku była świadkiem narodzin nowego pomysłu: komputera osobistego. Jego konstruktor, Amerykanin polskiego pochodzenia, Stephen Wozniak (wcześniej pracownik firmy elektronicznej ATARI) nadał mu przewrotną nazwę APPLE (jabłko). W ciągu zaledwie kilku lat mikrokomputery te zyskały wielki rozgłos w całych Stanach Zjednoczonych. Sprzedawane są również w Europie Zachodniej, Azji, Australii.

Mimo wciąż rosnącej konkurencji, zainteresowanie JABŁKIEM nie słabnie. Równie dynamicznie rozwija się przy tym wielki rynek oprogramowania oraz dodatkowego sprzętu. Wersja APPLE II posiada obecnie najbogatsze, spośród wszystkich komputerów osobistych, oprogramowanie, a także — wielostronne możliwości dalszej rozbudowy.

Powstają liczne kluby zrzeszające posiadaczy i zwolenników APPLE.

Znane są też liczne związane z firmą periodyki.

Istnieją również ciekawe projekty (niektóre już zrealizowane) wykorzystania mikrokomputera APPLE II w szkolnictwie. Na terenie RFN, w miejscowości Worms, gra on pierwszorzędną rolę w dużym ośrodku szkoleniowym, w skład którego wchodzi technikum, centrum informatyki, gimnazjum i szkoła zawodowa. Pod kierunkiem prof. A. Rissbergera, uczniowie zapoznają się z mikroelektroniką i zasadami informatyki. Podobne szkoły istnieją także w Szwajcarii.

W grudniu ubiegłego roku Musee d'Art Moderne w Paryżu gościł m.in. APPLE II na specjalnej wystawie poświęconej syntezie i twórczości artystycznej obrazów. W połączeniu z plotterem, dwiema kamerami oraz płytą graficzną APPLE przetwarzał i kreował ruchome obrazy.

Centralną częścią APPLE II jest 8-bitowy mikroprocesor 6502 z listą 56 rozkazów, trzynastoma sposobami adresowania, z jednym akumulatorem oraz obszarem adresowania — do 64 KB (16 KB ROM + 48 KB RAM). APPLE wyposażony został w klawiaturę ASCII z 52 klawiszami, w tym klawisze specjalne: ESC, CTRL, RESET, REPT. Do odczytu znaków wykorzystany jest monolityczny dekodery ROM MM5740.

Efekty dźwiękowe można realizować za pomocą zwartego wewnątrz obudowy ośmioomowego głośnika o mocy 0,5 W, sterowanego przez wzmacniacz Darlingtona.

APPLE II zasilany jest przez wysokiej klasy zasilacz wysokoczęstotliwościowy (1,023 MHz) o napięciach wejściowych od 107 do 132 V lub 214 do 264 V oraz napięciach wyjściowych: +5 V, +11,8 V, -5,2 V i -12 V. W zasilaczu zastosowano szereg układów stabilizujących.

Na ekranie monitora standardowo wyświetlane są 24 linie tekstu, po 40 znaków w każdej. Wydruk tekstu możliwy jest w trzech wariantach:

**normal** — białe litery na czarnym tle  
**invers** — czarne litery na białym tle  
**flash** — wariant migający.

APPLE posiada także możliwości graficzne:

— mała rozdzielczość, czyli 40×40 punktów + 4 wiersze tekstu, 15 kolorów; lub 280×192 punkty, 6 kolorów  
— duża rozdzielczość, czyli 280×160 punktów + 4 wiersze tekstu, 6 kolorów; lub 280×192 punktów, 6 kolorów

Pamięć zewnętrzną mogą stanowić:  
— zwykła pamięć kasetowa  
— dyskietki (ang. floppy disk) — 280 KB  
— dyski twarde (ang. hard disk) — 10 MB

Cena: APPLE II (bez dodatków) — ok. 990 dol.; APPLE II + monitor + stacja napędów do dyskietek (format 5¼") — ok. 1400 dol.

APPLE II dysponuje umiarkowaną dokładnością (9 cyfr). Nie są więc możliwe — często o kluczowym znaczeniu — obliczenia z tzw. podwójną dokładnością. Brak małych liter ogranicza możliwości redagowania tekstów. APPLE II nie przewiduje też możliwości mieszania tekstu z grafiką, a więc np.: opisu wykresów. Żadna wersja APPLE nie pozwala na bezpośrednie podłączenie go do telewizora. Ponadto fakt, iż zasilacz umieszczony został w bardzo bliskim sąsiedztwie głównej płytki, powoduje po dłuższym czasie pracy zbyt duże jej nagrzanie, co w rezultacie może prowadzić do przekłamań w programach.

Większość problemów rozwiązana została przez dodatkowe moduły sprzętowe wraz z towarzyszącym im oprogramowaniem. APPLE zyskał popularność nie tyle za sprawą zalet systemu, lecz również, a może przede wszystkim z powodu swoich nieprzeciętnych możliwości dalszej rozbudowy. Podwójna dokładność możliwa jest po wstawieniu płytki z mikroprocesorem Z80. Może on pracować zupełnie niezależnie lub współpracować z 6502.

Małe litery można wprowadzić w dwójaki sposób. Pierwszy polega na wmontowaniu tzw. **80-Z-CARD**. Karta ta „drukuję” (na monitorze) tekst w 80 kolumnach z dużymi i małymi literami oraz szereg innych znaków, niezbędnych przy redagowaniu tekstu. Drugi sposób to odpowiednie oprogramowanie pozwalające na wprowadzenie małych liter (programy typu: **SCREENWRITER**).

Program **HRTEXT** lub znacznie droższy **WORDSTAR** (ok. 400 dol.) prócz małych i dużych liter, zestawia tekst z grafiką. Uzyskiwane jest to przez wpisanie punktów tworzących napisy do pamięci obrazowej, tak jakby to była grafika. Stosunkowo łatwo jest więc wprowadzić alfabet grecki czy polski.

Podłączenie do dowolnego czarno-białego telewizora umożliwia dość prosta **UHF-CARD**. Z kolorowymi jest trochę gorzej. Kandydaci do współpracy z APPLE muszą pracować w systemie PAL. Potrzebna jest także specjalna płytka **PAL-CARD**. Podłączenie do kolorowego telewizora uwiódniecia niezmiernie bogaty świat grafiki APPLE. Nie najlepiej jest jednak z ostrością tekstu — często występują problemy z przypadkowym jego zabarwieniem.

Konstrukcja APPLE II pozwala na podłączenie od 8 do 16 dodatkowych płytek. I tu pojawia się problem; często bowiem niektóre z nich wykluczają się nawzajem.

Na potrzeby APPLE II opracowano trzy systemy operacyjne: **DOS**, **UDCSD** **Pascal**, **CP/M**. Dwa ostatnie umożliwiają kontakt z innymi mikrokomputerami.

W standardowej wersji oprogramowania występuje interpreter **INTERGERBASIC** lub **APPLESOFT**. Za pomocą tzw. **LANGUAGE CARDS** można korzystać z następujących języków: **PASCAL**, **LOGO**, **APLUS**, **FORTH**, **TRANSFORTH**, **FORTRAN**, **COBOL**, **TINY-C**, **GRAFORTH**.

**APPLESOFT** — jest pewną modyfikacją języka **BASIC**.

**APLUS** — zawiera w sobie **APPLESOFT** i stanowi jego znaczne rozszerzenie, jest to „pre-compiler”, umożliwiający programowanie strukturalne.

**FORTH** — język ten został napisany specjalnie na potrzeby mini- i mikrokomputerów. Dla **APPLE** istnieje w formie interpretera 10–100 razy szybszego niż **BASIC**. **TRANSFORTH** — napisany przez Paula Lutosa jest pewną modyfikacją języka **FORTH**; uwzględniono w nim specyfikę **APPLE**.

**GRAFORTH** — również przystosowany dla potrzeb **APPLE II**, łączy w sobie wszystkie zalety języka **FORTH**, jak i wykorzystanie grafiki dwu- i trójwymiarowej wraz z animacją oraz muzyką.

**TINY-C** — opracowany przez **BELL LABS** w formie interpretera, umożliwia programowanie strukturalne. (cdn)

**PIOTR TYMOCHOWICZ**

01-950 Warszawa  
ul. Renesansowa 5 m. 10  
tel. dom. 35-77-60

W dyspozycji autora znajdują się adresy przedstawicielstw firmy **APPLE** i klubów użytkowników. Dysponuje on również mikrokomputerem **APPLE II** i znaczną biblioteką oprogramowania; chętnie więc nawiąże kontakt z wszystkimi zainteresowanymi.

● Publikujemy kolejne adresy „agencur” **KUM-u**. Podane niżej osoby dysponują ankietami klubowymi, zbierają oferty i zapotrzebowania. **KOSZALIN**: Jan Marczak — 75-846, ul. Słowiańska 15; tel. d. 333-20, p. 277-91 w. 262. **GLIWICE**: Janusz Najzarek — 44-100, ul. Słowackiego 22 m. 5; tel. p. 31-80-21 w. 1875. **OSTROWIEC ŚWIĘTOKRZYSKI**: Romuald Redlich — 27-400, Osiedle Słoneczne 46/55; tel. p. 236-52 w. 3652. **OPOCZNO**: Krzysztof Surdyk — 26-300, ul. Partyzantów 55 m. 31; tel. d.33-50, p. 34-11 w. 201.

● Informacje o charakterze niekomercyjnym zamieszczamy bezpłatnie. Autorów większych opracowań prosimy o wcześniejsze uzgodnienie z nami tematu i formy prezentacji — coraz częściej bowiem kilka osób opisuje to samo.



— prowadzi  
**Andrzej J. Piotrowski**  
(tel. dom. 48-22-85)

# MULTIBUS-II — wieloprocessorowy system modułarny

W ubiegłym dziesięcioleciu największą rolę na rynkach światowych odgrywały dwa systemy modułarne: NIM<sup>1)</sup> (analogowy) i CAMAC (cyfrowy). Oba wywodzą się z atomistyki, znalazły jednak szerokie zastosowanie również w wielu innych dziedzinach. W okresie tym niemal każdego roku powstawały nowe systemy cyfrowe, pod względem technicznym lepsze od systemu CAMAC, jednakże nie różniące się na tyle od CAMACA, aby mogły wyprzeć go z pozycji standardu światowego.

Pojawienie się w 1981 roku (znowu w atomistyce) nowego systemu FASTBUS nie stanowiło konkurencji dla CAMACA, gdyż FASTBUS jest systemem o wyjątkowo dużej szybkości (70 MB/s) ograniczonym w zasadzie do zastosowań w obrębie fizyki wysokich energii, a zatem — o ograniczonym rynku światowym. Wprowadzenie systemu VME przez firmę MOTOROLA stanowiło następny krok. W ostatnich trzech latach z systemem tym wiązano pewne nadzieje, gdyż wiele ośrodków wprowadziło go do użytku, a kilka firm rozpoczęło produkcję i utworzył się rynek światowy o wielkości około 30 mln dolarów rocznie. Wydaje mi się jednak, że system VME w obecnej wersji nie ma szans na dalszą znaczną ekspansję.

W 1983 roku pojawiły się prawie równoległe dwa nowe opracowania: system P896.1 oraz MULTIBUS-II (P896.2). Pierwszy z nich został opracowany przez IEEE (Institute of Electrical and Electronics Engineers) przy udziale EDISG (European Distributed Intelligence Study Group). System MULTIBUS-II opracowała głównie firma INTEL, wykorzystując w dużej mierze prace prowadzone przez IEEE w ramach projektów P896, P959 i P796 (odpowiednio FUTURE BUS, SBX i MULTIBUS). Aczkolwiek oba te systemy są mniej więcej równorzędne technicznie, to jednak wiadomo już, że P896.1, nie mający poparcia znaczących producentów, nie zostanie wprowadzony na rynek światowy. Natomiast MULTIBUS-II z bazą produkcyjną firmy INTEL (i około 80 innych firm) jest już obecnie głównym kandydatem na przyszły standard, a przynajmniej — na pierwszą pozycję na rynku światowym.

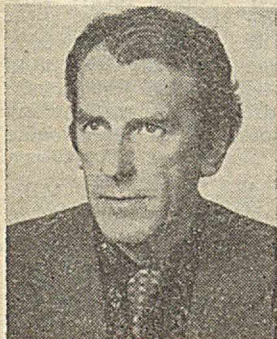
Niniejsze opracowanie ma na celu przedstawienie charakterystyki technicznej systemu MULTIBUS-II.

## CHARAKTERYSTYKA OGÓLNA

Powstanie MULTIBUS-II jest efektem wielu różnych tendencji przejawiających się w ubiegłych latach, związanych z rozwojem obwodów scalonych i elektroniki cyfrowej. Do głównych cech tego systemu należą:

● **wieloprocessorowość** — w stosunku do poprzednich systemów, MULTIBUS-II jest rozwiązaniem w pełni wieloprocessorowym, w którym już struktura arbitrażu jest zdecentralizowana

<sup>1)</sup> NIM — Nuclear Instrumentation Modules



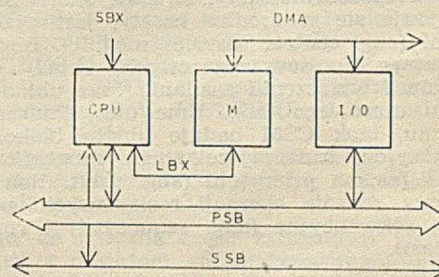
Doc. dr inż. ROMAN TRECHCIŃSKI do 1957 r. pracował w Instytucie Łączności, zajmując się zagadnieniami rozgłaszania przewodowego. W latach 1957–58 był zastępcą kierownika Polskiej Wyprawy Polarnej na Spitsbergen, a od 1959 r. pracuje w Instytucie Badań Jądrowych w Świerku (obecnie Instytut Problemów Jądrowych), gdzie zajmuje się problematyką modułarnych systemów cyfrowych — poprzednio systemu CAMAC.

● **struktura wielomagistralowa** — w systemie MULTIBUS-II występuje pięć różnych magistral, których organizacja logiczna jest zróżnicowana i dostosowana do wybranych grup zadań realizowanych przez te magistrale

● **optymalizacja konstrukcji mechanicznej** — zastosowano tzw. Eurokarty i złącza typu C96 (96-kontaktowe); doświadczenia wielu poprzednich rozwiązań pozwalają na wybór optymalnych wymiarów spośród kilku opcji, np.:

- wysokość — 233,35 mm (podwójna Eurokarta)
- głębokość — 220 mm
- szerokość — 20,32 mm

● **szybkość transmisji** dowolnie wybierana w zakresie do 40 MB/s dla magistrali równoległej i 48 MB/s dla magistrali lokalnej (rys. 1); maksymalna szybkość pozwala na realizację wielu zastosowań, nawet bardzo specyficznych, z dziedziny automatyzacji badań naukowych; mniejsze szybkości mogą być stosowane zależnie od konkretnych potrzeb



Rys. 1. Architektura systemu MULTIBUS-II  
oznaczenia: PSB — magistrala równoległa (ang. parallel system bus), SSB — magistrala szeregową (ang. serial system bus), LBX — magistrala lokalna (ang. local bus extension), SBX — magistrala uzupełniająca (ang. system bus extension), DMA — magistrala bezpośredniego dostępu, CPU — procesor, M — pamięć, I/O — wejście-wyjście

● **detekcja błędów** z bitem parzystości, w zasadzie — na każdy bajt przesyłany po magistrali; prawdopodobieństwo wystąpienia błędu zostało znacznie zmniejszone przez zastosowanie synchronizowanej transmisji asynchronicznej (ang. synchronous handshake); system może być stosowany przy silnych polach zakłócających

● **długość słowa danych** — 32, 24, 16 lub 8 bitów

● **adresowanie** 32-bitowe do komunikacji z pamięcią, 16-bitowe przy obsłudze wejścia-wyjścia i komunikacji wewnętrznej (ang. interconnect space), 8-bitowe do przesyłania komunikatów (ang. messages)

● **jednolite traktowanie przerw i zgłoszeń**

● **bogata organizacja logiczna części dotyczącej sterowania**: pozwala to na dostosowanie się do konkretnych potrzeb obsługiwanego procesu

● **względnie szybki arbitraż** (obsługa zgłoszeń) — trzy cykle zegarowe

● **pozostawienie dolnego złącza C96 do dyspozycji projektanta (użytkownika)**; pozwala to na stosowanie systemu w zestawach o dużej liczbie wejść i wyjść obiektowych

● **sygnały według logiki TTL.**

Podstawową cechą systemu MULTIBUS-II jest uniwersalność, tj. dostosowanie do bardzo różnorodnych wymagań użytkowych. Dlatego oczekuje się, że MULTIBUS-II szybko zdobędzie rynki światowe.

Architekturę systemu przedstawiono na rysunku 1. Magistrale PSB i SSB (równoległa i szeregowa) zajmują górne złącze C96. Magistrala LBX (lokalna) wykorzystuje dolne złącze, lecz nie może łączyć więcej niż sześć stanowisk. Magistrala SBX (uzupełniająca) stanowi połączenie pomiędzy tzw. komputerem jednopłytkowym SBC (ang. single board computer) a modulem rozszerzającym (SBX multimodule). W tym celu na płycie SBC jest umieszczone odpowiednie złącze. Jeżeli do SBC dodaje się płytkę multimodule, to taki układ zajmuje co najmniej dwa stanowiska w kasecie. Złącza do magistrali DMA (bezpośredniego dostępu) są umieszczane na przedniej stronie modułu. Po szczególne bloki w specyfikacji stosuje się nazwę agent) nie muszą mieć połączeń do wszystkich magistrali — zależy to od decyzji projektanta.

Specyfikacja MULTIBUS-II nie określa miejsca umieszczenia zasilacza. Zakłada się, że decyzję tę musi podjąć producent kasy i magistrali, np. w zależności od przyjętego sposobu wentylacji. Na ogół zasilacze umieszcza się nad lub pod kasetą. Stosowane są również zasilacze zajmujące kilka miejsc w kasecie. Jako napięcia zasilające przyjęto +5 V, +12 V i -12 V, +5 V z baterii. Nie sformułowano ograniczeń na maksymalne prądy. W jednym z istniejących rozwiązań przyjęto np. 70 A dla +5 V i 2 A dla ±12 V.

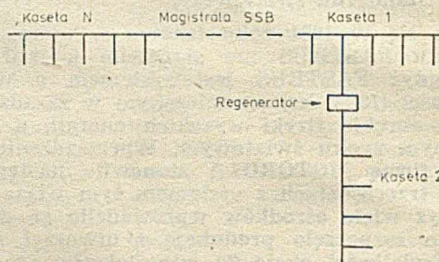
**MAGISTRALA RÓWNOLEGLA PSB**

Podstawowe parametry elektryczne i logiczne magistrali PSB podano powyżej. Magistrala równoległa łączy wszystkie stanowiska w kasecie. Stanowiska są numerowane od strony lewej w sposób nieciągły: 10, ..., 19, 0, ..., 9. Stanowisko zerowe, znajduje się w środku kasy, jest przeznaczone na blok CSM (ang. central services module). Blok ten zawiera generator impulsów zegarowych, układy działające przy włączeniu i wyłączeniu zasilania oraz układ przekroczenia limitu czasowego (ang. time out). Podczas inicjowania systemu, blok CSM nadaje innym blokom numery identyfikacyjne, oznaczające adres geograficzny (ang. cardslot id) i poziom priorytetu (ang. arbitration id). Wyróżnia się trzy rodzaje operacji rozkazowych: pojedyncze (ang. single), sekwencyjne (ang. sequential) i rozgłaszanie (ang. broadcast).

W przypadku jednoczesnego zgłoszenia się dwóch lub kilku bloków, uruchamiana jest procedura arbitrażu wykorzystująca linie BREQ (zgłoszenie) oraz ARB5—ARB0 (arbitraż). Istnieją dwie klasy priorytetu: ARB5=1 — priorytet wysoki i ARB5=0 — projekt normalny. Dalsze rozróżnienie priorytetu zapewnia numer identyfikacyjny (ang. arbitration id) wysyłany przez zgłaszające się bloki na linii ARB4—0. Blok o najwyższym priorytecie uzyskuje dostęp do magistrali (trzy cykle zegarowe — 300 ns), zaś pozostałe oczekują na zakończenie operacji i uzyskują dostęp w następnej kolejności, według numeru identyfikacyjnego priorytetu. Inne bloki o normalnej klasie priorytetu nie mogą się zgłaszać, dopóki wszystkie poprzednie zgłoszenia nie zostaną załatwione. Natomiast blok o wysokim priorytecie może zgłaszać się w dowolnej chwili i obejmie go procedura arbitrażu.

Takie rozwiązanie arbitrażu, a także inne cechy systemu MULTIBUS-II, jak duża szybkość, wieloprocesorowość i struktura wielomagistralowa — pozwoliły na uniknięcie odrębnego mechanizmu przerwań. Przerwania zachodzą identycznie jak zgłoszenia, przy czym operacja przerwania odbywa się w obszarze tzw. przekazywania komunikatów (ang. message space). W trakcie tej operacji przekazywane są informacje o rodzaju przerwania. Po uzyskaniu dostępu do magistrali przez jeden z bloków nadrzędnych (ang. master) następuje cykl transferowy (ang. transfer cycle) składający się z dwu faz (ang. request phase, reply phase). W pierwszej fazie po multipleksowanych liniach AD31—AD0 przekazuje się adres, a po liniach SC9—SC0 odpowiednie sygnały sterujące i informacyjne. W drugiej fazie adres zastępowany jest danymi, przy czym czynne są również linie sterujące, jednak znaczenie niektórych z nich jest inne niż w fazie pierwszej. Jeśli w trakcie cyklu transferowego zostanie stwierdzony błąd, wówczas występuje następny cykl (ang. exception cycle), podczas którego sterownik nadrzędny uzyskuje informacje o rodzaju błędu i uruchamiana jest procedura likwidacji błędu (ang. recovery phase).

Magistrala szeregowa jest objęta specyfikacją MULTI-BUS-II, jednakże firma INTEL uprzedziła zainteresowanych, że w roku 1984 możliwe jest jeszcze wprowadzenie zmian. Ponadto, już obecnie użytkownicy zgłaszają życzenie unifikacji magistrali szeregowej w skali światowej, gdyż może ona służyć nie tylko do transmisji wewnątrz jednej kasy, lecz także między kasetami, a nawet pomiędzy różnymi urządzeniami. Pełna unifikacja w skali światowej byłaby więc bardzo pożądana. Obecnie jednak istnieje kilka różnych rozwiązań, z których co najmniej dwa (VMS firmy MOTOROLA lub SSB MULTIBUS-II) mogą być brane pod uwagę jako podstawa przyszłego standardu. Obie wymienione magistrale (VMS i SSB) można zakwalifikować do klasy CSMA/CD (ang. carrier-sense multiple access with collision detection, dostęp wspólny z wykrywaniem kolizji). Główna różnica dotyczy sposobu usuwania kolizji. Czy unifikacja rzeczywiście zostanie osiągnięta — zależy raczej od względów pozatechnicznych (uzgodnienia między firmami MOTOROLA i INTEL).



Rys. 2. Struktura magistrali szeregowej

Magistrala SSB zajmuje dwa kontakty górnego złącza, do których dołączone są dwie linie oznaczone SDA i SDB. Strukturę magistrali szeregowej przedstawiono na rysunku 2. Można do niej dołączyć 32 bloki. Jeżeli w określonym zastosowaniu jest to niewystarczające, to poszczególne kasety można połączyć przez układ sprzęgający tzw. regeneratory (ang. repeater). Szybkość transmisji wynosi 2 Mb/s, a maksymalna długość magistrali — 10 m. Do detekcji błędów zastosowano 16-bitowe pole CRC (ang. cyclic redundancy check). Format transmisji na magistrali szeregowej odpowiada formatowi magistrali równoległej PSB w obszarze tzw. przekazywania komunikatów (ang. message space). Sygnały przesyłane są w systemie kodowania bifazowego (ang. Manchester coding), według zasady przedstawionej w tabeli:

Możliwe stany magistrali szeregowej

Linia	SDA	SDB
Wolna	1	1
Logiczne 1	1	0
Logiczne 0	0	1
Kolizja	0	0

**MAGISTRALA LOKALNA LBX**

Magistrala lokalna jest przeznaczona wyłącznie do komunikacji procesorów (bloki typu master) z pamięciami wykonanymi jako bloki danej kasy i może łączyć powyżej sześć stanowisk (zajmuje dolne złącza kasy). Stosowane są magistrale obejmujące np. cztery stanowiska. Na magistrali lokalnej mogą pracować najwyżej dwa bloki typu master (lub tzw. requester). Jeden z nich jest zawsze uprzywilejowany (ang. primary), a drugi podporządkowany (ang. secondary) głównemu. Dzięki temu unika się w ogóle stosowania procedury arbitrażu.

Maksymalna szybkość transmisji jest równa 48 MB/s (częstotliwość zegarowa 12 MHz). Pojemność pamięci może wynosić 64 MB, a dane są przekazywane jako słowa 8-, 16- i 32-bitowe. Różni się dwa pola adresowe (ang. memory i interconnect spaces), odpowiadające formatowi PSB. W jednej kasecie może istnieć więcej niż jedna magistrala lokalna, o ile jest to potrzebne.



## MAGISTRALA BEZPOŚREDNIEGO DOSTĘPU DMA

Magistrala DMA, zewnętrzna w stosunku do kasety MULTIBUS-II, obejmuje 60 linii (pary skręcane lub płaskie wieloprzewodowe kable) i jest wyprowadzona przez specjalne złącze 60-kontaktowe, umieszczone na przedniej stronie bloku. Pełni podobne funkcje jak tzw. magistrala GPIB (ang. general purpose interface bus, objęta normą IEC-625) w aparaturze pomiarowej. Jest przeznaczona głównie do komunikacji międzysystemowej oraz — urządzeń wejścia-wyjścia z pamięcią. Umożliwia transmisję z szybkością 8 MB/s na odległość 15 m. Do magistrali można dołączyć 16 urządzeń, które mogą należeć do trzech grup:

- supervisor (tylko jeden)
- controller
- talker-listener.

Dane są przekazywane w postaci słów 8- i 16-bitowych, w dwóch obszarach adresowych, pamięci i wejścia-wyjścia — po 16 MB każdy.

## MAGISTRALA UZUPEŁNIAJĄCA SBX

Wprowadzenie tej magistrali ma na celu stworzenie możliwości wykorzystania tzw. modułów rozszerzających (ang. SBX multimodules), objętych zaleceniem IEEE P959. Moduły te umożliwiają rozszerzenie funkcji komputerów jednopłytkowych SBC (ang. single board computer), które stosuje się jako bloki nadrzędne (ang. master) w systemie MULTIBUS-II. Płytkę multimodule łączy się z płytką master przez złącza wielokontaktowe — tak, że stanowią one razem blok o podwójnej szerokości. Magistrala SBX umożliwia transmisję słów 8- i 16-bitowych, do czego stosuje się odpowiednio złącza 32- lub 44-kontaktowe. Przewiduje się dwie klasy transmisji — z bezpośrednim dostępem i bez.

\* \* \*

Na zakończenie warto zastanowić się, czy można w najbliższym czasie oczekiwać pojawienia się propozycji jeszcze innego standardu systemu modularnego, który mógłby stać się konkurencyjny w stosunku do MULTIBUS-II. Doświadczenia ostatnich lat wskazują niezbicie, że źródłem standardu światowego może być tylko duża organizacja mająca oparcie w przemyśle. W chwili obecnej istnieją tylko dwie takie organizacje, związane z firmami INTEL i MOTOROLA. MOTOROLA jest obecnie zaangażowana w rozwój standardu VME i nie wiem nic o przygotowywaniu ewentualnej następnej propozycji. Jeżeli nawet taka próba zostanie podjęta teraz, to sam standard pojawi się za kilka lat, co może być już zbyt późno na wyeliminowanie MULTIBUS-II, który do tego czasu opanuje dużą część rynku światowego. Jednocześnie MULTIBUS-II — z tech-

nicznego punktu widzenia — jest niewątpliwie systemem najbardziej nowoczesnym. Wynika stąd wniosek, że jeżeli już teraz należy dokonać wyboru, a taki pogląd jest dosyć powszechny wśród fachowców, to nie ma lepszego wyboru jak — MULTIBUS-II.

Pozytywne przyjęcie systemu MULTIBUS-II na świecie spowoduje, że w najbliższych latach należy oczekiwać pojawienia się (poza mikroprocesorami) specjalistycznych obwodów scalonych przeznaczonych do wykonywania poszczególnych funkcji, jak np.:

- wymiana informacji po magistrali — (ang. transceiver and protocol chip)
- arbitraż (ang. arbitration chip)
- sterowanie magistralą szeregową (ang. serial bus chip) itp.

Jednakże już teraz można projektować układy MULTIBUS-II na bazie istniejących obwodów. W tym przypadku przy wersji 16-bitowej strata miejsca na płytkach wynosi ok. 10% powierzchni, a więc względnie niewiele.

Architektura systemu VME jest bardzo zbliżona do MULTIBUS-II. Wprawdzie w specyfikacji VME nie występują odpowiedniki magistrali SBX i DMA, są to jednak magistrale zewnętrzne w stosunku do magistrali kasety i można je traktować jako uzupełniające. Magistrala SSB systemu MULTIBUS-II ma swój odpowiednik w magistrali szeregowej VME (tzw. VMS Bus), a w 1984 roku firma MOTOROLA zaanonsowała wprowadzenie magistrali lokalnej VMX, która jest odpowiednikiem magistrali LBX. Tak więc główne różnice pomiędzy VME i MULTIBUS-II można scharakteryzować następująco:

- MULTIBUS-II zapewnia większą powierzchnię płytki (głębokość modułu 220 mm, zamiast 160)
- w MULTIBUS-II obie główne magistrale mieszczą się w jednym rzędzie złącz (VME obejmuje dwa rzędy)
- inna baza elementowa (MOTOROLA i INTEL)
- w MULTIBUS-II stosuje się arbitraż wieloprocessorowy, a w VME — centralny (blok typu arbiter)
- w MULTIBUS-II stosuje się synchronizowane operacje „handshake”, a w VME — typowy „handshake”
- szybkość transmisji w MULTIBUS-II wynosi 40 MB/s, a w VME — 20 MB/s
- w VME stosowane są tzw. połączenia łańcuchowe (ang. daisy chain) do wyróżnienia stanowiska, a w MULTIBUS-II takich linii nie ma.

Przyjmując, że znaczna część zastosowań wymaga licznych połączeń z obiektem automatyzowanym, a takie połączenia mogą być prawidłowo zaprojektowane tylko przez złącza na tylnej stronie modułu, należy stwierdzić, że system VME mógłby być zastosowany w tych przypadkach tylko w wersji 16-bitowej. Pozostaje więc do rozstrzygnięcia czy systemy 16-bitowe można obecnie uznać za perspektywiczne. Dzisiejsze tendencje wskazują raczej na rozwiązanie 32-bitowe.

## Zasady prenumeraty

Zamówienia i przedpłaty na prenumeratę **INFORMATYKI** przyjmuje Zakład Kolportażu Wydawnictwa **NOT SIGMA**. Adres pocztowy: Wydawnictwo **NOT SIGMA** — Zakład Kolportażu, 00-950 Warszawa, skr. poczt. 1004. Konto bankowe: 1036-7490-139-11, III O/M NBP w Warszawie.

**JEDNOSTKI GOSPODARKI USPOŁECZNIONEJ, INSTYTUCJE I ORGANIZACJE** przesyłają zamówienia (w 1 egz.) zawierające: tytuł czasopisma, liczbę zamawianych egzemplarzy, okres prenumeraty i pełny adres zamawiającego z kodem pocztowym, oddział i nazwę banku z numerem konta bankowego zamawiającego oraz (ewentualnie) adres odbiorców, którzy na zlecenie i koszt zamawiającego mają egzemplarze otrzymywać.

Warunkiem realizacji zamówienia jest równoczesne dokonanie odpowiedniej wpłaty na ww. konto Wydawnictwa **NOT SIGMA**.

Za prenumeratę nie wystawiane są rachunki i nie potwierdzane są salda. Prenumeratory zbiorowi proszeni są o podawanie na dowodach wpłat (przelewach) znaku kancelaryjnego zamówienia, którego dotyczy wpłata.

Dopisując na zamówieniu **PRENUMERATA STAŁA**, zamawiający (tylko prenumeratory zbiorowi) nie będą musieli corocznie ponawiać zamówienia, a jedynie dokonywać przedpłaty według aktualnie obowiązujących cen. Wydawnictwo przekazywać będzie co roku potwierdzenie kontynuacji prenumeraty.

**PRENUMERATORZY INDYWIDUALNI** dokonują wpłaty przekazem NBP na ww. konto, pod powyższym adresem, podając na odwrocie odcinka dla adresata-posiadacza rachunku: tytuł czaso-

pisma, liczbę zamawianych egzemplarzy oraz okres prenumeraty.

Do **PRENUMERATY ULGOWEJ** upoważnieni są członkowie stowarzyszeń naukowo-technicznych **NOT**, studenci, uczniowie szkół zawodowych. Warunkiem jej uzyskania jest poświadczenie blankietu przekazu NBP dla nabywcy indywidualnego (na odcinku dla adresata) przez właściwe stowarzyszenie **NOT**, wyższą uczelnię lub szkołę zawodową.

Zamówienia i wpłaty przyjmowane są na okresy kwartalne, półroczne i roczne w terminach:

- do 15 listopada — na I kwartał, I półrocze i cały rok następny
- do 28 lutego — na II, III i IV kwartał
- do 31 maja — na IV kwartał i II półrocze
- do 31 sierpnia — na IV kwartał.

Uwaga: Przy podawaniu kodu pocztowego i numeru konta bankowego obowiązuje bardzo czytelne pismo. Prenumerata nie wymaga specjalnego przekazu z czerwonym paskiem; wystarczy zwykły przekaz bankowy.

Prenumerata normalna: kwartalna — 225 zł, półroczna — 450 zł, roczna — 900 zł. Prenumerata ulgowa: kwartalna — 120 zł, półroczna — 240 zł. Prenumerata ze zleceniem wysyłki za granicę jest dwukrotnie droższa.

Dodatkowych informacji o prenumeracie udziela: Zakład Kolportażu, tel. 40-00-21 w. 293, 299 oraz 40-35-89. Egzemplarze archiwalne można nabywać w Klubie Prasy i Informacji Technicznej w Warszawie, ul. Mazowiecka 12, tel. 27-43-65. Zamówienia na egzemplarze archiwalne należy kierować pod adresem Zakładu Kolportażu.

# I/A. Struktura i modularność programów

Niniejszy zestaw pytań<sup>1)</sup> nie stanowi jakiegokolwiek standardu egzaminacyjnego i dlatego stosowanie go do sprawdzania wiedzy innej osoby — poza samym sobą — stanowi świadome pogwałcenie intencji ACM i zastrzeżonych praw autorskich tego stowarzyszenia.

**PYTANIA**

1. Co — ze względu na modyfikowalność programu — jest najważniejszym wyróżnikiem prawidłowych rozmiarów modułu programowego?

- a) nieprzekraczanie jednego arkusza kodowego
- b) nieprzekraczanie pełnej strony pamięci wirtualnej
- c) zredukowanie modułu do prostej i łatwej do zrozumienia funkcji
- d) mieszczanie się w maksymalnym bloku dyskowym

2. Rozważmy poniższy podprogram, zapisany w pewnym języku proceduralnym:

```
MODULE SILNIA (N)
RETURN (IF N<3 THEN N ELSE
        SILNIA (N-1) * N) ;
END SILNIA;
```

Które z podanych pojęć najlepiej scharakteryzowałoby taki program?

- a) iteracja
- b) rekursja
- c) dwuwymiar
- d) struktura

3. Trzema podstawowymi elementami konstrukcyjnymi w programowaniu strukturalnym są:

- a) sekwencja, rekursja, iteracja
- b) wywołanie, zamknięcie, selekcja
- c) rozmiar, wywołanie, format
- d) sekwencja, selekcja, iteracja

4. Oto fragment pewnego programu:

```
I = 1;
CASE I x 3 OF
1: I = 1;
2: I = 2;
3: I = 4;
END CASE;
```

Jaką wartość będzie miała zmienna I po wykonaniu tego fragmentu programu?

- a) 1; b) 2; c) 3; d) 4

5. Przeanalizujemy następujący fragment pewnego programu:

```
PROCEDURE PODST (A,B,C,D)
REAL A,B,C,D;
B = A+A;
D = A+C;
RETURN;
END;
...
...
Y=1; Y=2; Z=7;
CALL PODST (X,X,X+Y,Z);
PRINT Z;
```

Załóżmy, że przekazywanie parametrów odbywałoby się tutaj metodą odsyłaczową (ang. call-by-reference) — niekiedy zwaną także pozycyjną (ang. call-by-location) lub adresową (ang. call-by-address). Jaka wartość zostałaby wydrukowana jako „Z”?

<sup>1)</sup> Communications of the A.C.M., Vol. 19 (No. 5, May 1976) p. 231. Stowarzyszenie ACM zezwoliło na przedruk Self-Assessment Procedure I (Copyright 1976 by Association for Computing Machinery) do celów niekomercyjnych

- a) 4; b) 5; c) 6; d) 7

6. Można jednak założyć, że przekazywanie parametrów w przykładzie z pytania 5 odbywałoby się metodą bezpośrednią, tzn. „poprzez wartość” (ang. call-by-value). Co wtedy byłoby wydrukowane jako wartość „Z”?

- a) 4; b) 5; c) 6; d) 7

7. Przekazywanie parametrów w przykładzie z pytania 5, można założyć, mogłoby także odbywać się metodą identyfikacyjną, czyli „poprzez nazwy” (ang. call-by-name). Jaka wartość zostałaby wydrukowana w takim przypadku jako „Z”?

- a) 4; b) 5; c) 6; d) 7

8. Rozpatrzmy następujący fragment programu, napisanego w pewnym języku proceduralnym:

```
...
A = 6;
LOOP: DO INDEX=N TO 10 BY 1;
A=A+1;
END LOOP;
PRINT A;
...
```

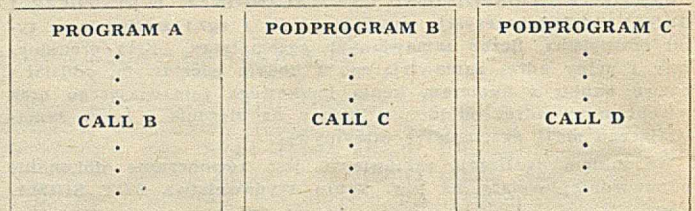
Załóżmy, że użyta tutaj instrukcja DO działa w ten sposób, że warunek badany jest na początku pętli — a na wejście do pętli podano wartość N=12. Jaka wartość zostanie wydrukowana w „A”?

- a) 6; b) 7; c) 8; d) 9

9. Pewien przemieszczalny (ang. relocatable) program ładujący (ang. loader) wymaga, aby plik wynikowy zawierał określone informacje pomocnicze. Którą z wymienionych wersji takich wymogów można uznać w tym przypadku za najmniej prawdopodobną?

- a) zewnętrzne informacje symboliczne — zarówno dotyczące wszystkich symboli zdefiniowanych wewnątrz programu wynikowego, odwołania do których występują poza nim, jak i dotyczące wszystkich symboli, do których odwołuje się sam program wynikowy, ale które zdefiniowane są poza nim
- b) informacje tekstowe — obejmujące przemieszczalny kod wynikowy, program źródłowy i długości segmentów
- c) informacje tablicowe — określające tablicę symboli, zawierającą nazwy i atrybuty wszystkich symboli użytych w programie źródłowym
- d) informacje przemieszczalne — złożone z odwołań do tych wszystkich miejsc kodu wynikowego, których zawartość zależy od adresów przydzielonych programowi

10. Wyobraźmy sobie trzy moduły programowe:



Każdy z tych modułów został skompilowany oddzielnie i zawiera odwołanie do innego modułu zewnętrznego, przy czym moduł D jest zewnętrznym dla A, B i C. Załóżmy, że moduły te poddano działaniu programu łączącego (ang. linkage-editor). Czy można określić ile wówczas zostanie utworzonych odwołań do nazw zewnętrznych?

- a) 0; b) 1; c) 2; d) 3

11. Niechaj dany będzie fragment modułu programu, zapisany w języku symbolicznym z użyciem notacji dziesiątkowej. Załóżmy, że adresem bazowym dla tego modułu

jest 0, którą to wartość umieszczono w polu +50 w trakcie ładowania modułu. Jaka będzie zawartość pola 65 po wykonaniu podanego fragmentu?

Adres	Operacja	Argument
10	Zeruj akumulator i dodaj zawartość pola	14
11	Do akumulatora dodaj zawartość pola	14
12	Zawartość akumulatora zapamiętaj w polu	15
13	Przenieś do pola	5
14	Zdefiniuj stałą o wartości	50
15	Zdefiniuj obszar o wymiarze	1
..	...	..

a) 1; b) 50; c) 100; d) 200

## ROZWIĄZANIA

'c-11 'c-01 'c-6 'v-8 'c-1 'd-9 'q-2 'd-1 'q-2 'c-1

## KOMENTARZE

- ad 1. Yohe J.: An Overview of Programming Practices. Computing Surveys, Vol. 8 (No. 4, December 1974), p. 223  
Aron J.: The Program Development Process. Addison-Wesley, 1975, p. 99-102
- ad 2. Elson M.: Concepts of Programming Languages. Science Research Associates, p. 177-187
- ad 3. Nicholls J.: The Structure and Design of Programming Languages. Addison-Wesley, 1975, p. 17-18
- ad 4. Pratt T.: Programming Languages. Design and Implementation. Prentice-Hall, p. 143-144
- ad 5. Gries D.: Compiler Construction for Digital Computers. Wiley, p. 188  
Nicholls — ibid. p. 503-505  
Elson — ibid. p. 77-79
- ad 6. Gries — ibid. p. 188  
Nicholls — ibid. p. 502-503  
Elson — ibid.

- ad 7. Gries — ibid. p. 190-191  
Nicholls — ibid. p. 499-502  
Elson — ibid.
- ad 8. Ralston A.: Introduction to Programming and Computer Science. McGraw-Hill, p. 320  
Iverson K.: A Programming Language. Wiley, p. 6
- ad 9. Donovan J.: Systems Programming. McGraw-Hill, p. 160-164
- ad 10. Freeman P.: Software Systems Principles. Science Research Associates, p. 471-475
- ad 11. Donovan — ibid. p. 156-160  
Chapin N.: Computers: A Systems Approach. Van Nostrand, p. 237-238.

\* \* \*

Po przestudiowaniu pytań i osobistym skonfrontowaniu własnych odpowiedzi z sugerowanymi rozwiązaniami — oraz ewentualnym sięgnięciem do literatury źródłowej, do której podane komentarze bibliograficzne stanowią jedynie wstępne naprowadzenie — należy zastanowić się nad użytecznością przedstawionego samotestu I/A. Zdaniem testodawców — ACM Committee on Self-Assessment — osoba zainteresowana niniejszym samotestem może uznać go za bezwzględnie przydatny dla siebie, o ile tylko:

- uświadomi sobie istnienie pewnych pojęć, uprzednio nie znanych lub niezbyt dobrze rozumianych,
- rozszerzy swą wiedzę o głębsze rozumienie pojęć istotnych dla wykonywanej pracy lub osobistych zainteresowań.

Od siebie dodajemy, iż mając na względzie trudność uzyskania w naszym kraju sugerowanej przez ACM literatury — głównie zresztą podręcznikowej — zapewne nasi Czytelnicy zechcą podzielić się z Redakcją propozycjami bardziej dostępnych źródeł, a także innymi uwagami, uściśleniami terminologicznymi itp. skorygowaniami przedstawionej próbki samotestu.

Adaptowali z angielskiego  
ADAM B. EMPACHER  
LUDWIK J. ROSSOWSKI

EO/432/K/84

**Z KRAJU**

## Rada ds. Zastosowań Środków Techniki Obliczeniowej

### Posiedzenie Rady

Rada ds. Zastosowań Środków Techniki Obliczeniowej zebrała się 1 i 2 marca 1984 w Moskwie na swym VIII posiedzeniu. Uczestniczyli w nim przedstawiciele wszystkich ośmiu krajów członkowskich, a obrady prowadził nowy przewodniczący Rady — Arnold Romanow, zastępca przewodniczącego Państwowego Komitetu Nauki i Techniki ZSRR. On też rozpoczął posiedzenie, przedstawiając sprawozdanie z ubiegłorocznej działalności. Następnie przedstawiono sprawozdania części krajowych oraz organów roboczych. Po wysłuchaniu sprawozdań, Rada podsumowała wykonanie prac przewidzianych do zakończenia w ubiegłym roku. Stwierdzono, że na 32 zaplanowane przedsięwzięcia wykonano 19; oceniono przyczyny niezakończenia pozostałych, urealnialiśmy jednocześnie odpowiednie terminy.

Wśród zakończonych prac znalazł się generator słownika zautomatyzo-

wanych systemów sterowania procesami technologicznymi dla maszyn serii SM (Instytut Systemów Sterowania, Katowice) oraz program do komputerowo wspomaganego nauczania chemii fizycznej PHYCHEX-JS (Politechnika Rzeszowska). Rada podkreśliła, że podpisano już cztery kontrakty w celu realizacji zadań zawartych w planie współpracy i że przygotowuje się do podpisania dzieł następných. Jeden z kontraktów, zawarty pomiędzy organizacjami bułgarskimi i radzieckimi, dotyczy wspólnego opracowania pakietu programów dla zautomatyzowanego magazynu. Świadczy to o tym, że metoda ta, będąca ciągle jeszcze nowością, znalazła już zrozumienie i poparcie krajów członkowskich. Pojawienie się metod handlowych w obrocie oprogramowaniem spowodowało również wystąpienie Rady do Komisji Międzyrządowej z prośbą o opracowanie norm ustalania cen na programy.

W dalszej części obrad dokonano niezbędnych korekt Jednolitego Planu Współpracy na lata 1984-1985. Usunięto z niego niektóre tematy, które nie wywołały zainteresowania innych krajów, niż zgłaszające, bądź utraciły źródła finansowania. Rada zatwierdziła także Plan Organizacyjny, określający spotkania w bieżącym roku, przypominając jednocześnie o obowiązujących terminach zawiadamiania o miejscu spotkania i przesyłania związanych z nim materiałów.

Rozpoczęto też prace przygotowawcze do opracowania Jednolitego Planu Współpracy na lata 1986-1990. Postanowiono, że części krajowe Rady przedstawiają swoje propozycje do września 1984 wraz z określeniem podstawowych kierunków prac do roku 1995. W trakcie obrad omawiano strukturę Rady i uznano za celowe przedłużenie istnienia dotychczasowych grup roboczych oraz powołanie Tymczasowej Grupy Roboczej Zautomatyzowanych Systemów Zarządzania, co ma zapewnić lepszą koordynację prac prowadzonych w tej dziedzinie. Zatwierdzono także na stanowisko przewodniczącego sekcji SAPR — Witalija Bojko — szefa części ZSRR tej sekcji.

W dalszej części zatwierdzono wstępny program kolejnego posiedze-

nia Rady, ustalając, że odbędzie się ono w styczniu 1985 r. w Polsce. Również w naszym kraju, pod koniec bieżącego roku, przeprowadzona zostanie narada sekretarzy krajowych części Rady.

Podsumowując obrady, przewodniczący Rady określił problemy, które będą głównym nurtem jej prac:

- miejsce i rola informatyki w realizacji kompleksowych programów automatyzacji i elektronizacji krajów RWPG
- znaczenie informatyki dla gospodarki narodowej, z podkreśleniem szczególnej roli automatyzacji procesów

technologicznych (robotyzacja, elastyczne systemy produkcyjne itp.)

- poprawa efektywności tworzenia systemów informatycznych (normalizacja, unifikacja, rozwój metod, problem wyceny prac prowadzonych na zasadach kontraktowych)
- poprawa efektywności wykorzystania prac (dotychczas wykorzystuje się zaledwie ok. 20% wykonanych systemów)
- wypracowanie skuteczniejszych metod współpracy dwu- i wielostronnej
- skoncentrowanie sił na głównych kierunkach działania

● szczególna rola prac prowadzonych przez grupę przygotowania kadr

● szersze wykorzystanie Centrum Koordynacyjnego MKETO do usprawnienia prac Rady, a w szczególności do usprawnienia obiegu informacji.

Na zakończenie reprezentanci Węgier zaprosili do udziału w wystawie „Mikrokomputery w gospodarce narodowej” (11—14.09.1984, Budapeszt) i w seminarium „Informatyka w administracji państwowej” (12—13.09.1984).

Oprac. Mks  
na podstawie materiałów RZ STO

## ZE ŚWIATA

### Fotony w komputerze

Wszystko wskazuje na to, że na przekór stałemu postępowi w elektronice, komputery osiągnęły już górną granicę szybkości, wyznaczoną prędkością elektronów. Czy fotony, najszybsze cząstki, zastąpią elektrony w obwodach? Badania nad materiałami nieliniowymi pozwalają rozważać możliwość konstruowania komputerów optycznych, których szybkość tysiącrotnie przewyższy możliwości maszyn elektronicznych. Ponadto urządzenia optyczne pozwalają na jednoczesne przetwarzanie dużej liczby sygnałów, otwierając tym samym drogę ku nowym architekturom systemów cyfrowych.

W takim to entuzjastycznym tonie spotkać można na ten temat liczne publikacje w czasopismach zagranicznych. Zdecydowaliśmy się jedną z nich udostępnić Czytelnikom, mimo szeregu wątpliwości co do sposobu wyjaśnienia przez autora tekstu źródłowego zasad działania transfazera, a także przesadnego wyolbrzymienia zalet niesprawdzonej jeszcze technologii. (Red.)

Minęło już dwadzieścia lat od chwili, kiedy równocześnie narodziły się laser i optyka nieliniowa. Podczas gdy pierwsze z odkryć cieszyło się niezwykłym rozgłosem i wciąż fascynowało, świat nauki nowymi zastosowaniami, optyka nieliniowa wzbudziwszy początkowo nadzieje zastosowania jej w informatyce lub telekomunikacji — natrafiła wkrótce na trudności w znalezieniu materiałów wykazujących odpowiednie właściwości.

Okolo roku 1975 osiągnięcia optyki nieliniowej znalazły się ponownie w polu zainteresowania informatyki, kiedy to zaczęto myśleć o zastąpieniu urządzeń elektronicznych — optycznymi, gdzie przesyłanie sygnałów odbywa się już nie za pośrednictwem prądu elektrycznego, lecz fali świetlnej, a ściślej — wiązki laserowej.

Światło, jak wszystkie fale elektromagnetyczne, posiadając prędkość największą z możliwych, stanowi idealny nośnik informacji dla ultraszybkich komputerów. Generalnie fotonika (franc. photonique) ma trzy istotne zalety w stosunku do elektroniki:

- te same możliwości funkcjonalne sprzętu osiąga się przy dużo mniejszej objętości (rzędu 1000 razy)
- przetwarzanie odbywa się setki razy szybciej
- duża liczba sygnałów może być przetwarzana jednocześnie bez obawy wzajemnej interferencji.

Aby skonstruować komputer w którym miejsce elektronów zajmą fotony należało znaleźć urządzenie dwustabilne, optyczny odpowiednik tranzystora. Właśnie skojarzenie lasera i materiałów nieliniowych pozwoliło na opracowanie tranzystora optycznego czyli transfazera (franc. transphaseur).

Tranzystor optyczny jest urządzeniem działającym na zasadzie interferometru Pérot'a-Fabry'ego zastosowanego w optyce nieliniowej. Interferometr składa się z dwóch częściowo przepuszczających światło luster, ustawionych równolegle jedno za drugim w pewnej odległości, oraz z zawartej między nimi przestrzeni, gdzie zachodzi interferencja fali przechodzącej przez interferometr i fali odbitej od powierzchni drugiego zwierciadła.

W naszym przypadku przestrzeń pomiędzy zwierciadłami wypełnia prostokątny kryształ materiału nieliniowego, zaś rolę zwierciadeł pełnią powierzchnie wypolerowanej przedniej i tylnej ścianki kryształu. Wiązka laserowa o częstotliwości właściwej dla danego kryształu jest kierowana na przednią powierzchnię interferometru i częściowo odbijana, powiedzmy w 90%; pozostałe 10% tworzy wiązkę załamana, która z kolei zostaje w 90% (9% wartości wiązki wejściowej) odbita na powierzchni tylnej. Powierzchnia tylna emituje zatem na zewnątrz

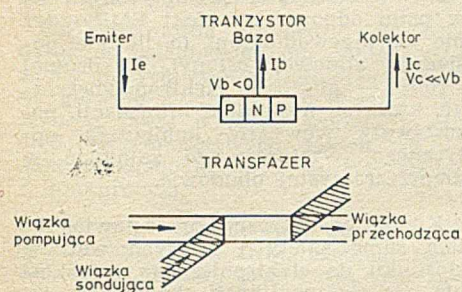
wiązkę o natężeniu 1% wiązki wejściowej.

W rzeczywistości, wiązki wewnątrz przyrządu interferują, tworząc falę, której amplituda w każdym punkcie stanowi sumę amplitud składowych. Dla niektórych relacji pomiędzy długością fali i długością urządzenia interferencja jest zjawiskiem destruktywnym — następuje stłumienie fal, natężenie światła wewnątrz interferometru jest prawie zerowe i natężenie wiązki przechodzącej jest nieznaczne. Dla innych relacji — wiązki, załamana i powracająca, wzmacniają się wzajemnie i wiązka przechodząca ma natężenie równe natężeniu wiązki wejściowej; wewnątrz interferometru działa w tym wypadku jak rezonator optyczny.

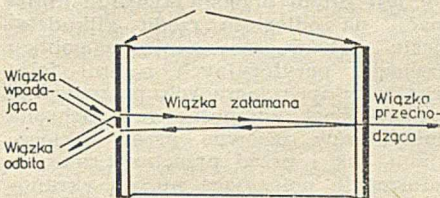
Tu właśnie wkracza do akcji nieliniowość: kiedy materiał tworzący wnętrze interferometru jest nieliniowy, wystarczy zmieniać natężenie wiązki wpadającej, aby uzyskać zmianę długości wiązek interferujących — załamanej i odbitej, osiągając tą drogą zmianę wspomnianych relacji długości. W ten sposób, bez modyfikacji długości interferometru ani częstotliwości wiązki laserowej, mała zmiana natężenia wiązki wejściowej będzie mogła pociągnąć za sobą raptowny wzrost natężenia wiązki przechodzącej. Natężenie wiązki zmieni się więc gwałtownie od wartości bliskiej zeru do wartości natężenia wiązki wpada-

jącej. Nazwa transfazer pochodzi stąd, że pozwala on na zmianę fazy przechodzącego przezeń światła.

mi żadne zjawiska związane z interferencją. Poza tym optyka nieliniowa powinna pozwolić na realizację systemów o więcej niż dwóch stanach stabilnych. Transfazer może mieć więcej stanów rezonansu, odpowiadających różnym natężeniom wiązki wpadającej.



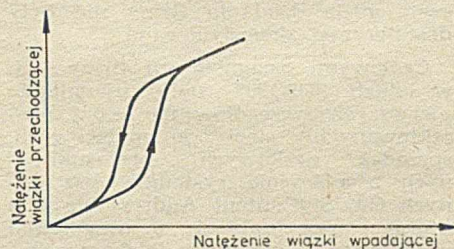
Rys. 1. Prądom emitera, bazy i kolektora w tranzystorze odpowiadają trzy wiązki transfazera. Wiązki pompująca i sondująca są wiązkami laserowymi idealnie spójnymi, skierowanymi na przednią powierzchnię interferometru Pérot'a-Fabry'ego. Wiązka pompująca jest silna, a jej natężenie jest takie, by natężenie wiązki przechodzącej było równe zero. Natężenie słabszej wiązki sondującej może być regulowane w ten sposób, by powodować zmianę wiązki przechodzącej od najmniejszej wartości natężenia (dla wiązki sondującej równej zero) do wartości maksymalnej. Dwóm wartościom natężenia wiązki przechodzącej można przypisać wartości logiczne „0” i „1”. Powyższe urządzenie zostało opracowane przez E. Abrahama, C. Seaton'a i D. Smith'a z Uniwersytetu Heriot'a-Watt'a w Edynburgu



Rys. 2. Interferometr Pérot'a-Fabry'ego składa się z prostokątnej płytki, z przezroczystego materiału, umieszczonej pomiędzy dwoma częściowo przepuszczającymi światło zwierciadłami. Wpadająca wiązka laserowa jest częściowo odbijana od powierzchni przedniej interferometru. Wiązka załamana, częściowo odbita od powierzchni tylnej, dzieli się na wiązkę powracającą i wiązkę przechodzącą. Wiązki powracająca i załamana interferują wewnątrz przyrządu. Jeśli interferencja jest destruktywna, natężenie światła pomiędzy zwierciadłami jest praktycznie zerowe, tak samo jak natężenie wiązki przechodzącej. Jeżeli przeciwnie — interferencja jest konstruktywna, natężenie światła wewnątrz jest bardzo duże, wiązka przechodząca ma zatem natężenie równe natężeniu wiązki wpadającej. W transfazerze wewnątrz interferometru stanowi kryształ nieliniowy, którego wypolerowane ścianki tworzą półprzepuszczalne zwierciadła

Zmiana natężenia wiązki przechodzącej, jako funkcja natężenia wiązki wpadającej, tworzy cykl histerezy charakterystyczny dla urządzeń dwustabilnych. Jest to krzywa zmian podobna do krzywej opisującej prąd w kolektorze tranzystora. W praktyce, zamiast zmieniać natężenie wiązki wpadającej, dekomponuje się ją na wiązkę pompującą (o stałym natężeniu i taką, by natężenie wiązki wysyłanej było równe zero) oraz wiązkę sondującą, która poddana superpozycji z wiązką pompującą powoduje zmianę natężenia wiązki przechodzącej od wartości minimalnej (dla wiązki sondującej równej zero) do maksymalnej.

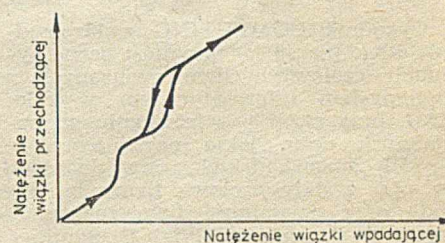
Główną cechą komputera optycznego będzie jednak jego olbrzymia prędkość działania, tysiąc razy większa od najmniejszych dokonań elektronicznej techniki obliczeniowej. Będzie też możliwa większa gęstość zapisu informacji, a małe zużycie energii jest także zaletą nie bez znaczenia.



Rys. 3. Dwustabilność jest wspólną cechą tranzystora i transfazera. Natężenie wiązki przechodzącej przez transfazer jest funkcją natężenia wiązki wpadającej. Natężenie wiązki przechodzącej zmienia się w sposób określony krzywą podobną do tej, która opisuje tranzystor: początkowo posiada wartość bliską zero, następnie — wraz ze wzrostem natężenia wiązki wpadającej — przechodzi raptownie na poziom równy mu wartości, by na nim pozostać. Kiedy natężenie wiązki wpadającej maleje, natężenie wiązki przechodzącej maleje również, jednakże z pewnym przesunięciem spowodowanym przez światło pozostające w kryształach

We Francji prowadzone są obecnie badania w laboratorium fizyki ciała stałego na uniwersytecie Piotra i Marii Curie w Paryżu. Kieruje nimi profesor Minko Balkanski. Równoległe prace wykonuje się na Politechnice w Strasburgu. Przygotowuje się podstawy konstrukcji komputerów optycznych, zarówno pod względem organizacji, jak i materiałów. W ostatnich latach udało się uzyskać materiały nieliniowe przejawiające pożądane właściwości w normalnej temperaturze. Dotychczas dla otrzymania efektu nieliniowości wymagane były temperatury bardzo niskie. Z chwilą urzeczywistnienia transfazera, technologia komputerów optycznych nie powinna nastęrczać większych trudności, można bowiem wzorować się na rozwiązaniach elektronicznych, zastępując połączenia elektryczne światłowodami.

Każda wiązka przechodząca przez przezroczysty materiał ulega spowolnieniu w zależności od właściwości tego materiału. Źródłem spowolnienia jest zjawisko refrakcji: współczynnik załamania ośrodka jest równy stosunkowi prędkości światła w próżni do jego prędkości w danym ośrodku. Częstotliwość fali (kolor) nie ulega zmianie.



Rys. 4. Wylączną cechą transfazera jest dwustabilność wielokrotna. Wynika ona z faktu, że skoro natężenie wiązki wpadającej będzie większe, to zmieni się współczynnik załamania materiału nieliniowego, a — co za tym idzie — długość fali wewnątrz kryształu. Można więc zwiększając natężenie wiązki wpadającej, osiągnąć sukcesywnie kolejne stany rezonansu.

Zasadniczą zaletą komputera optycznego jest jego szybkość, która pozwoli mu przekroczyć teoretyczne ograniczenia elektroniki. Z chwilą kiedy współczesne superkomputery osiągnęły skrócenie czasu cyklu rozkazowego z dziesiątek ns do 1 ns ( $10^{-9}$  s), wyczerpały swoje możliwości w tym zakresie. Tranzystor optyczny uzyskuje czas przełączania rzędu 1 ps ( $10^{-12}$  s). Biorąc pod uwagę pozostałe właściwości fotoniki można liczyć, że czas cyklu rozkazowego uda się zmniejszyć o kilka rzędów wielkości. Ponadto tranzystor optyczny oferuje niezwykłą możliwość jednoczesnego przetwarzania wielu sygnałów. W określonym stanie kwantowym może znajdować się dowolna liczba fotonów, co wynika ze statystyki Bosego-Einsteina, zaś w wypadku elektronów podlegających statystyce Fermiego-Diraca jest to niemożliwe.

Tak więc dla fali o danej częstotliwości jej długość w określonym materiale jest zdeterminowana przez współczynnik załamania tego materiału. Zwykle materiały przezroczyste, liniowe, mają współczynnik załamania stały, niezależny od natężenia wiązki wpadającej, które jest proporcjonalne do natężenia wiązki przechodzącej. Materiały nieliniowe, w których współczynnik załamania zależy od natężenia wiązki wpadającej — to przede wszystkim krystaliczne półprzewodniki, np. antymonek indu i arsenek galu. Zmiana natężenia światła wystarcza wówczas do spowodowania zmiany długości fali wewnątrz kryształu.

W rezultacie przez transfazer może przejść jednocześnie kilka wiązek, przy czym nie zachodzą pomiędzy ni-

Technologie są bardziej wyrafinowane, a komputery, tak jak wszystkie systemy elektroniczne, stają się coraz wrażliwsze na niestabilność napięcia, fale elektromagnetyczne i inne zakłócenia. Eksplozja jądrowa w wysokich warstwach atmosfery powoduje istny chaos elektromagnetyczny na dużym

obszarze. Jest to tak zwany efekt EMP (Electromagnetic Pulse) lub inaczej „radioflash”. Mamy wtedy do czynienia z promieniowaniem elektromagnetycznym o krótkiej długości fali, o częstotliwościach w przedziale 1—100 MHz. Badania wykazały, że promieniowanie to jest w stanie zniszczyć układ scalony w czasie krótszym od 1  $\mu$ s, gdyż powstające pod jego wpływem prądy powodują stopnienie metalicznych połączeń. Promieniowanie może też być przyczyną przekłamań na torze przesyłania informacji, powodując zamianę „0” na „1” lub odwrotnie, co technicy określają jako „latch up”. Promieniowanie neutronowe, towarzyszące eksplozji jądrowej, ma zdolność penetracji struktury krystalicznej półprzewodnika, przy czym następuje zmiana tej struktury poprzez liczne dyslokacje, co pociąga za sobą niekorzystną zmianę właściwości.

Komputery nie tylko są wrażliwe na zakłócenia, ale same także stanowią ich źródło. Każde urządzenie elektroniczne wytwarza wokół siebie zmienne pole elektromagnetyczne. W rezultacie poszczególne elementy systemu informatycznego zachowują się jak nadajniki radiowe o częstotliwości emitowanej fali od 2 do 500 MHz. Możliwe jest zatem przechwytywanie informacji na odległość rzędu kilkudziesięciu metrów, szczególnie w pobliżu urządzeń peryferyjnych i linii transmisyjnych łączących poszczególne ogniwa sieci komputerowej. Ochrona komputerów zarówno przed zakłóceniami, jak i przed przechwytywaniem informacji, polegająca np. na ekranowaniu wszystkich elementów systemu, jest trudna, a niejednokrotnie — niemożliwa.

Tymczasem komputer optyczny może być penetrowany przez różnego

rodzaju sygnały bez wplywu na sygnał właściwy przetwarzany w urządzeniu. Ponadto światłowody stosowane w komputerach optycznych są w pełni odporne na efekt EMP, gdyż nie są przewodnikami (o ile nie posiadają stalowej osłony). Z drugiej strony — aby fale elektromagnetyczne, które są nośnikami informacji, pozostały wewnątrz komputera optycznego, wystarczy go zamknąć w nieprzezroczystej obudowie.

Komputer optyczny przedstawia zatem same zalety i żadnych wad w stosunku do elektronicznego, co skłania do intensyfikacji badań w tej dziedzinie.

Opracował **MAREK BIENKO**

na podstawie artykułu Claire Remy: „Des photons dans l'ordinateur”, MICRO-SYSTEMES, grudzień 1983

## Przemysł informatyczny 1983 w ocenie brytyjskiej

Tygodnik COMPUTING z 15 grudnia ub.r. (nr 49/83) przyniósł ocenę sytuacji czołowych firm związanych z przemysłem informatycznym w roku 1983, przy czym w odróżnieniu od podobnych ocen, jakie uzyskujemy ze źródeł amerykańskich, uwzględnione w nim zostały czołowe firmy brytyjskie oraz działalność firm zagranicznych na terenie Wielkiej Brytanii.

Druga w kolejności światowa potęga komputerowa DIGITAL EQUIPMENT CORPORATION (DEC) od półtora roku przechodzi dotkliwy kryzys i żaden doradca finansowy nie zalecałby obecnie inwestowania w tę firmę. Jej światowe obroty rosną już znacznie wolniej niż w latach poprzednich. Koszty badań i przeciągających się opracowań w dziedzinie komputerów osobistych, czołowego wyrobu firmy — systemu VAX oraz pamięci dyskowych, znacznie obniżyły zyski firmy. W ciągu ostatnich pięciu kwartałów notowano kolejne zmniejszenie zysku o 36, 38, 38, 29 i 72%. W Wielkiej Brytanii sytuacja tej firmy jest nieco lepsza.

W 1984 r. DEC niewątpliwie będzie starał się odrobić straty. Obecnie można stwierdzić, że prawie cała działalność firmy koncentruje się na systemie VAX. Wstrzymano opracowywanie komputerów 36-bitowych DEC-SYSTEM 10 i 20, natomiast rok ten przyniesie opóźnione opracowanie nowego modelu o nazwie SUPER-VAX. Wśród małych systemów pojawi się VAX zrealizowany w jednej koscie, a więc ten system będzie dostarczany w pełnej gamie możliwości obliczeniowych. Opracowana zostanie również nowa generacja komputerów osobistych, która zastąpi m.in. uniwersalny komputer biurowy RAINBOW

(„Tęcza”) oraz przeznaczony dla specjalistów PROFESSIONAL. Po opracowaniu obu tych modeli DEC miał kłopot z terminową realizacją dostaw, co znacznie utrudniło dalsze zwiększenie sprzedaży. Należy jednak zauważyć, że w 1984 r. amerykańskie firmy TYMSHARE i SYSTEMS CONCEPTS będą już sprzedawać konkurencyjne mikrokomputery 36-bitowe. Nie wiadomo również, czy oczekujący na SUPER-VAX nie zwrócą się w stronę systemów IBM i kompatybilnych z nimi wyrobów.

Największą troską DEC jest spodziewany w bieżącym roku znaczny wpływ użytkowników jego wyrobów, zwłaszcza w kategorii mniejszych maszyn, gdzie działania firmy nie były dostatecznie skoordynowane.

Czołowymi osiągnięciami firmy ICL w latach 1982/1983 było ujawnienie pierwszych szczegółów na temat projektowanych sieci komputerowych, sprzedaż dyskowych, przeszukiwanych asocjacyjnie systemów baz danych CAFS (Content Addressable File Store), oraz zabiegi o utrzymanie współpracy z gigantem telekomunikacyjnym BRITISH TELECOM (BT).

W dziedzinie sieci firma przyjęła system ETHERNET i prowadzi już szereg ośrodków doświadczalnych sieci tego typu. Jednocześnie w roku 1984 pojawiają się pierwsze kostki ETHERNET. Jednakże korzystać z tego systemu będą mogli tylko użytkownicy nowego systemu operacyjnego VME, co jest tym dziwniejsze, że najnowsze komputery opracowywane na lata dziewięćdziesiąte nadal wykorzystują stary system operacyjny GEORGE-3/DME.

Systemy bazy danych CAFS stały się już standardowym wyposażeniem

komputerów rodziny 2900. W przypadku komputerów korzystających z systemu operacyjnego DME koszt CAFS wynosił 250 tys. funtów, natomiast wszystkie jednostki z systemem VME wyposażone są już w taki system bazy danych.

Innym rozbudowywanym i bieżąco ulepszanym systemem ICL jest DAP (Distributed Array Processor), który w pewnych zastosowaniach może działać 30—40 razy szybciej niż komputer CRAY-1. Warto podkreślić, że ICL również odstępuje swoje podstawowe oprogramowanie specjalistycznym firmom handlowym.

Pod koniec 1984 r. ma pojawić się komputer o nazwie DM/1, będący owocem współpracy ICL z japońską firmą FUJITSU, a w rok później jeszcze większy model o nazwie ESTRIEL, które to systemy w przyszłości stopniowo zastąpią komputery rodziny 2900. W kategorii systemów średniej wielkości obniżono ceny komputerów rodziny ME 29, jednakże obniżka ta nie obejmuje oprogramowania systemów bazy danych.

Drugim w kolejności źródłem zysków ICL jest minikomputerowy SYSTEM 25, tym bardziej, że dotychczas nie osiągnięto większych sukcesów w sprzedaży systemów DRS (Distributed Resource Systems — Systemy o Rozłożonych Zasobach). Dopiero dołączenie systemu operacyjnego UNIX powinno zwiększyć zainteresowanie tymi systemami. Natomiast inna maszyna ICL wyposażona w UNIX. — PERQ sprawiła wiele kłopotów i musi być gruntownie przepracowana, jeśli ma się w ogóle utrzymać na rynku.

Współpraca ICL z firmami wytwarzającymi oprogramowanie nie jest duża, wynosi bowiem zaledwie 11 mln funtów na 850 mln łącznych obrotów, ale rozwija się ona pomyślnie dając nie tylko wartościowe produkty gotowe, ale i silny bodziec do rozwinięcia własnych prac w tej dziedzinie.

Ostatnio obserwuje się zmniejszenie aktywności ICL na rynku USA, mimo nominalnego wzrostu zamówień w

1983. Na rok bieżący przewiduje się podjęcie wysiłków dla poprawy sytuacji, m.in. wprowadzając do sprzedaży wspomniane maszyny DM/1.

Ogólnie można stwierdzić, że ICL dążąc do osiągnięcia poprzedniego poziomu dużych zysków zamierza oprzeć się głównie na dostawach dużych maszyn, a także dokonać szeregu posunięć oszczędnościowych, takich jak redukcja personelu, oraz wzmocnić walkę z konkurencją.

IBM w przeciwieństwie do omówionych oraz innych firm, osiągnął bardzo dobre wyniki, przy czym tym razem działalność firmy w Europie pozostawała nieco w tyle w stosunku do całości obrotów. Bardziej niż w ubiegłych latach IBM posługiwał się firmami pośredniczącymi, zarówno przy rozprowadzaniu komputerów osobistych jak i przy wyposażaniu swoich wyrobów w oprogramowanie opracowane przez firmy specjalistyczne. Ponadto zawarto szereg umów z firmami o nieco innym charakterze, jak np. z ROLM w celu zwiększenia udziału IBM na rynku sprzętu biurowego czy z producentem podzespołów INTEL. Wiele podobnych umów zawarł oddział IBM w Japonii.

W roku 1983 IBM wprowadził na rynek szereg nowych maszyn oraz usprawnień dotychczasowych modeli. SYSTEM 36 został powitany entuzjastycznie zarówno przez użytkowników, którzy woleli czekać na ten system niż zmieniać producenta, jak i wytwórców oprogramowania. Innym wydarzeniem roku było wprowadzenie przez firmę dwu nowych modeli rodziny 4300: 4361 — uniwersalnego i 4381 — przeznaczonej do zastosowań naukowych i projektowych.

W 1983 r. zmieniono również szereg postanowień dotyczących dzierżawy sprzętu — utworzone zostało własne specjalistyczne przedsiębiorstwo dzierżawy sprzętu IBM, a na firmy pośredniczące nałożono pewne ograniczenia.

W dziedzinie telekomunikacji można mówić ostatnio o pewnej liberalizacji. Dostosowując się do reguł walki konkurencyjnej BT zmieniła swą strukturę organizacyjną i oferowała lepszą jakość usług. Ponadto firma utworzyła nowy oddział pod nazwą MERLIN, specjalizujący się w sprzęcie automatyzacji biur. Firmy konkurencyjne mimo współzawodnictwa BT dostarczyły jednak wielu prywatnym odbiorcom automatycznych łącznic telefonicznych.

Trudniej wejść na brytyjski rynek telekomunikacyjny firmom zagranicznym. Próbowano to zrobić kanadyjska NORTHERN TELECOM, ale wówczas BT prawie podwoiła swe zamówienia na SYSTEM X (który będzie dominował w najbliższych latach w brytyjskich łącznicach lokalnych) i tym samym wyeliminowała przeciwników. Na froncie wewnętrznym konkurencji BT, jak np. MERCURY, ponieśli również porażki, a działające w ramach BT związki zawodowe udało się pozyskać do walki przeciwko rządowym planom reprivatyzacji przedsiębiorstwa.

Firmy produkujące układy scalone miały bardzo dobry rok (najlepszy od 1979), chociaż zaczynał się on wcale nie tak optymistycznie — było wiele zwolnień z pracy, zamrożenie uposażeń i obawa przed zalewem tanimi elementami japońskimi. W drugim kwartale zaczęły jednak rosnąć zamówienia, co niektórzy producenci zlekceważyli. W połowie roku stosunek zamówień do zrealizowanych dostaw osiągnął bardzo dużą wartość. Nadal nie podjęto jednak niezbędnych inwestycji, co przyczyniło się do braku podaży podstawowych elementów oraz półrocznych czasów oczekiwania na dostawę. Na pierwszym miejscu należy tu wymienić dynamiczne pamięci o pojemności 64 K bitów, standardowe elementy 74LS i nową generację procesorów 16-bitowych o dużym stopniu scalenia. W roku 1983 pojawiły się pierwsze procesory 32-bitowe, których produkcja rozpocznie się w roku bieżącym. Należałoby wymienić tu układy firmy INMOS. Natomiast na rynku mikroprocesorów 16-bitowych, który wciąż nie jest jeszcze zbyt duży i nadal powiększa się, dominują INTEL i MOTOROLA. INTEL zdobył poparcie IBM, który wybrał procesor 8088 do swego komputera osobistego, będącego ogromnym sukcesem rynkowym, a ponadto zakupił już 14% akcji INTELA z zamiarem zwiększenia tego udziału do 30%. Czołowym produktem INTELA jest procesor 80186, oparty na modelu 8086, ale z wielu dodatkowymi funkcjami, co pozwala zmniejszyć wyposażenie towarzyszące. Firma MOTOROLA opracowała rodzinę mikroprocesorów 16-bitowych opartą na elementach 68000, a ZILOG 8090 i NATIONAL SEMICONDUKTOR 16000 mają nadal znaczne zamówienia.

Na rynku pamięci pozycję dominującą osiągnęli Japończycy, ale inne firmy ostro bronią swoich interesów. Stowarzyszenia Przemysłu Półprzewodnikowego USA wezwało rząd do ochrony prawnej, a brytyjska firma INMOS sprzedała wszystkie swoje wyroby, ale nie jest w stanie szybko rozszerzyć produkcji i uruchomić masowego wytwarzania nowych opracowań przygotowanych w USA.

Jeśli chodzi o oprogramowanie, to zaczyna ono odgrywać coraz większą rolę. Wiąże się to z wstępującymi możliwościami mikrokomputerów stwarzających rozległy zakres nowych zastosowań. Takie pakietowe oprogramowanie funkcjonalne znacznie przyspiesza rozwój techniki informacyjnej. Jednocześnie burzliwy rozwój informatyki rodzi pilne potrzeby programowych powiązań między dużymi maszynami i mikrokomputerami.

W Wielkiej Brytanii firma BT i Ministerstwo Obrony wycofały swe poparcie finansowe dla prac nad językiem ADA, co prawdopodobnie doprowadzi w tym zakresie do zwiększenia zależności od USA.

Wielu komentatorów przepowiada wielki rozwój systemu operacyjnego UNIX w 1984 roku. Szereg firm zajmujących się oprogramowaniem opracowało lub jest w trakcie opracowywania „Unixowych” wersji oferowanych pakietów zastosowań. System ten

ma rzeczywiście wielkie możliwości, ale nie będzie na szeroko stosowany, dopóki nie przyjmą go główni producenci sprzętu.

Uważa się, że UNIX może stać się standardowym systemem operacyjnym, pozwalającym twórcom oprogramowania wytwarzać programy dla całego rynku, a nie jego małych części. Możliwość przenoszenia oprogramowania staje się więc coraz istotniejszym czynnikiem dalszego rozszerzenia rynku zbytu.

Największym wydarzeniem roku 1983 na rynku mikrokomputerowym był wspomniany już sukces komputera osobistego PC firmy IBM. W okresie krótszym od dwóch lat IBM przejął 25% rynku i wyprzedził firmę APPLE na liście producentów małych systemów. Wynik ten został osiągnięty nie dzięki przewadze parametrów technicznych wspomnianego wyrobu, ale przez fakt, że był to po prostu wyrób IBM, który stał się standardem. Wyraża się to m.in. w tym, że jego system operacyjny PC DOS stał się podstawą opracowania wielu producentów oprogramowania. Inni wytwórcy, łącznie z APPLE, muszą więc przyjąć kompatybilność z IBM. Przyjęcie systemu operacyjnego PC DOS przez IBM, od firmy MICROSOFT, uczyniło niemal identyczny system tej firmy — MSDOS wielkim sukcesem handlowym, ograniczając jednocześnie zasięg systemu CP/M80 firmy DIGITAL RESEARCH.

Pierwszy raz w tak krótkim czasie sytuacja rynkowa wywołała tyle poważnych konsekwencji dla różnych firm. Niektóre z nich znalazły się w kłopotach finansowych, a inwestorzy wpadli w panikę. Konkurując z IBM wytwórcy systemów mikrokomputerowych zabiegają u producentów oprogramowania o opracowanie rozwiązań dla specjalizowanych zastosowań.

Firma APPLE szeroko reklamowała swój system LISA, który ze względu na wysoką cenę jeszcze nie rozpoznał się zbyt, a wielu potencjalnych użytkowników nadal go wypróbowało. Ale dzięki LIZIE rozpoznał się urządzenie zwane „myszką”, służące do wskazywania wybranych obszarów na ekranie monitora i stosowanie oprogramowania scalonego. Również stosowanie „okienek” stało się popularne w nowych pakietach oprogramowania, opracowanych dla innych maszyn.

Obawy przed „inwazją” producentów japońskich okazały się przesadne. Ich udział w światowym rynku małych maszyn wynosi zaledwie 4%, a więc nie stanowi na razie zagrożenia dla konkurentów. Natomiast walka toczona za pomocą obniżek cen spowodowała, że największe korzyści z komputerów domowych odnieśli użytkownicy. Wśród producentów może jedynie COMMODORE i SINCLAIR wyszły obronną ręką, chociaż ta ostatnia miała kłopoty w USA. Niektóre jednak firmy, jak np. TEXAS INSTRUMENTS zostały na skutek strat całkowicie wyeliminowane z rynku.

Opracował  
JAN RYŻKO

## Elementarz mikroprocesorowy

Na naszym rynku wydawniczym, całkowicie ogołoconym z książek o mikroprocesorach (moim zdaniem — głównie z winy wydawców, co mogę udowodnić), pojawiła się wreszcie nowa książka pt. „Układy mikroprocesorowe”, autorstwa Piotra Misiurewicza<sup>1)</sup>. Ponieważ książki tego rodzaju pojawiają się u nas z częstotliwością jednej na rok — warto odnotować jej wydanie.

Celem książki jest — jak stwierdza autor — „przedstawienie architektury, list rozkazów oraz zagadnień związanych z programowaniem współczesnych systemów mikroprocesorowych, a także sposobów sprzęgania tych systemów z urządzeniami zewnętrznymi”. Ma ona być czymś pośrednim między przystępną encyklopedią o mikroprocesorach a omówieniem bardziej złożonych zagadnień programowania mikrokomputerów.

Ponad połowę jej zawartości (trzy rozdziały) poświęcono na omówienie trzech typów mikroprocesorów, które układają się w wyraźną linię rozwojową. W trzech kolejnych rozdziałach przedstawiono systematyczny wykład wiedzy o mikroprocesorach — od najbardziej popularnego w Polsce mikroprocesora 8080, przez bardziej nowoczesny i wszechstronny, lecz mniej znany — Z80, do najbardziej zaawansowanego, choć zupełnie nie stosowanego w kraju, 16-bitowego mikroprocesora 8086. W wymienionych rozdziałach, na tle architektury poszczególnych mikroprocesorów, omówiono ich listy rozkazów oraz zasady pracy w systemach, a przy tej okazji — wiele podstawowych układów pomocniczych, a także — oprogramowanie najbardziej elementarnych funkcji, jak np. operacje arytmetyczne, konwersje danych, obsługa przerwań itp.

Rozdział dotyczący mikroprocesora 8080 jest dość nudny, choć to nie wina autora. Po prostu wszystkie przedstawione w nim fakty są na ogół dobrze znane; niemniej nie zaszkodzi je sobie utrwalić. Natomiast krótki rozdział o Z80 (mógłby być obszerniejszy) jest pierwszym w literaturze polskiej (por. A. J. Piotrowski, Systemy mikroprocesorowe nr 11, 1983, PIE) tak szczegółowym omówieniem listy rozkazów tego mikroprocesora, co jest znaczną zaletą książki. Brak tu trochę dokładniejszego omówienia większej liczby oryginalnych układów pomocniczych firmy ZILOG. Rozdział o mikroprocesorze 8086 można traktować jako minimalny wstęp do prezentacji mikroprocesorów 16-bitowych — i to jest również bardzo cenne. Przedstawiono w nim spójny pogląd na możliwości konstruowania i programowania zestawów (por. J. Grabowski, INFORMATYKA, nr 3—12, 1983).

Gdybym mógł coś dodać do tej części książki, to omówiłbym jeszcze bardziej szczegółowo (podał więcej przykładów) konkretne tryby adresowania, gdyż umiejętność posługiwania się nimi jest ważniejsza dla programowania, niż znajomość listy rozkazów. Brakuje mi też bardziej zdecydowanej krytyki omawianych mikroprocesorów, tj. przedstawienia ich ograniczeń, a szczególnie — tych funkcji, których przy ich użyciu nie można efektywnie zrealizować.

Druga część książki — rozdział pt. „Sprzęganie mikroprocesora z urządzeniami zewnętrznymi” — powinna być jej najciekawszym fragmentem. Po krótkim przedstawieniu ogólnych metod współpracy mikroprocesora z urządzeniami zewnętrznymi, omówiono tu sprzęganie szeregu typowych urządzeń występujących w zestawach, jak np. przetworniki analogowo-cyfrowe, klawiatury i wyświetlacze, dalekopis, a także odpowiednie układy LSI służące tym celom: układ transmisji szeregowej, czasomierz, układ bezpośredniego dostępu itp. Brak jest, natomiast, szczegółów dotyczących

współpracy z magnetofonem, monitorem ekranowym, a szczególnie — z dyskiem elastycznym.

Czy treść rozdziału spełnia oczekiwania Czytelnika? Tak, ale nie każdego. Nie jest to bowiem książka dla specjalistów, choć i dla nich będzie dobrym przypomnieniem, a nade wszystko — usystematyzowaniem posiadanych wiadomości. Jest to przede wszystkim doskonała lektura dla tych inżynierów, którzy muszą uczyć się podstaw techniki mikroprocesorowej, gdyż nie mieli jej na studiach, ani nie zetknęli się z nią pełniej w dotychczasowej praktyce zawodowej.

Rozdział o mikrokomputerach jednocukładowych (INTEL 8048) należałoby z pewnością rozbudować ponad obecnie zajmowane 15 stron książki. Jak zapowiada CEMI, odpowiednie układy będą produkowane w Polsce, ich dobra znajomość będzie z pewnością korzystna dla rozwoju tej techniki (por. artykuł A. Rakowskiego i A. Rosińskiego, str. 2). Oczywiście, przyjęcie nazwy **mikroprocesory jednocukłowe** jest nieporozumieniem. Choć jestem przekonany, że nikt jej nie będzie używał — po co mącić ludziom w głowach?

Drugi rozdział tej części, a zarazem ostatni rozdział książki, dotyczy systemów wieloprocusorowych i wielokomputerowych. Należy go traktować jako zasygnalizowanie tej problematyki. Choć autor ma świadomość, że nie jest to wyczerpujące jej potraktowanie, w kontekście całej książki takie ujęcie tematyki całkowicie wystarczy. Trzeba jednak pamiętać, że najbardziej, złożone jest nie tyle sprzęganie, co programowanie systemów wieloprocusorowych, w czym nie ma jeszcze wielu doświadczeń, zresztą nie tylko w Polsce.

Podsumowując — jest to pierwsza polska książka o mikroprocesorach napisana przez wykładowcę akademickiego (a można się spodziewać, że w najbliższych latach nie będzie ich więcej), a to ma określone konsekwencje. Takiej jasności wykładu próżno szukać u innych autorów. Z tego też względu, mimo inżynierskiego poziomu materiału, należałoby ją polecać przede wszystkim uczniom szkół średnich, gdyby ta tematyka była tam wykładana.

Choć autor boleje nad tym, że nie udało się w książce przedstawić zastosowań mikroprocesorów, jest to jej absolutnie niepotrzebne. Uwzględnienie choćby niewielu zastosowań spowodowałoby rozproszczenie wysiłku autora i uwagi Czytelników, przez co książka straciłaby na wartości. Zagadnieniom tym powinny być poświęcone oddzielnie, bardziej szczegółowe monografie. Przyszłe wydania tej książki, a nie wątpię, że takie być muszą, powinny uwzględnić rozwój tych dziedzin, które obecnie potraktowano raczej marginesowo, a dotyczy to szczególnie systemów wieloprocusorowych.

Pewien zarzut, jaki można postawić, choć może to nadmierna drobiazgowość, dotyczy zbyt dużej liczby rysunków — jest ich 279, a więc rysunek przypada prawie na każdą stronę. To o wiele za dużo, jak na moją zdolność percepcji. Rysunki mogą być (i są w tej książce) znakomitą ilustracją tekstu, ale przy tej ich liczbie czytelność książki znacznie się pogarsza. Kto potrafi je wyselekcjonować? Przy ostrzejszych kryteriach, połowa zapewne tych rysunków byłaby do pominięcia, szczególnie jeśli są to schematy blokowe. W porównywalnej objętościowo książce Ch. A. Titusa i in. o mikroprocesorach 16-bitowych (niestety, nie podjęto starań o jej przetłumaczenie), jest 181 rysunków, w tym 18 zdjęć i 43 programy, a więc ostatecznie — rysunków o połowę mniej.

Na zakończenie kilka uwag do Wydawcy. Znam metody pracy tej Redakcji i odpowiadają mi one. Nie sądzę, jednak, aby dobrzy redaktorzy nie mogli być jeszcze lepsi. A będą lepsi, jeżeli nie dopuszczą do „manipulowania na bitach” (str. 31) i operowania na nich (str. 29), a także — „operowania na danych” (str. 94, 107), „na zawartości” (str. 44) i „na ciągach” (str. 133, 158). Jeżeli już manipulujemy i operujemy, to „czym”, a nie „na czym”. Nie znam też przyczyny, która nakazuje pisanie skrótów we-wy dużą literą (We-Wy). Nie ma to żadnego uzasadnienia w języku polskim, a więc jest błędem. Usiłuję też dociec, co to jest „sygnał strobu statusu” (str. 49), a zwrot „okres czasu” (str. 46) przesyłam do wykorzystania w kabarecie. Te trochę złośliwe uwagi nie zmieniają jednak wysokiej oceny dla redakcyjnej pracy nad książką — to naprawdę trudna praca.

JANUSZ ZALEWSKI

<sup>1)</sup> Piotr Misiurewicz: Układy mikroprocesorowe — struktury i programowanie. Seria: Układy i systemy elektroniczne. Wydawnictwa Naukowo-Techniczne, Warszawa, 1983. Wydanie 1, str. 324, nakład 20 000 egz.



## O najczęstszych błędach w terminologii mikrokomputerowej (1)

W „mikroprocesorowym” numerze INFORMATYKI trudno powstrzymać się od uwag na temat poprawności terminologii stosowanej przez specjalistów z tej dziedziny. O ile w publikacjach książkowych na ogół bardzo starannie podchodzi się do spraw terminologicznych, to poziom wydawnictw zawodowych i prac instytutowych jest pod tym względem zatrważający. „Starsze” i „młodsze” bity, „skale integracji”, „operandy”, „kompatybilność” i wiele innych haseł łącznie tworzą bełkot doskonale utrudniający zrozumienie treści.

Sytuację pogarsza fakt, że do tej pory nie było dobrych wzorów. Jedynym, który może polecić uwadze Czytelników, jest wydrukowany w INFORMATYCE nr 1, 2/1983 „Słowniczek mikroprocesorowy” M. T. Jankowskiego, choć już nie jego podstawa — „Słowniczek mikrokomputerowy”, opublikowany w numerach 5, 6—7 i 8—9 ELEKTRONIKI, w 1981 roku. Inne opracowania słownikowe, np. cytowane w INFORMATYCE nr 2/1981 lub zbliżony do nich słowniczek A. Wiśniewskiego<sup>1)</sup>, mają bardzo roboczy charakter i nie przedstawiają zbyt dużej wartości normatywnej, gdyż stanowią jedynie wielojęzyczne zbiory odpowiedników bez określeń definicyjnych poszczególnych terminów.

Nawet tak podstawowy termin jak large scale integration jest prawie zawsze niewłaściwie tłumaczony jako duża skala integracji. Mimo, że large znaczy duży, scale — skala, a integration — integracja, large scale integration — to nie duża skala integracji, bo takie sformułowanie nic nie znaczy (nawet, jeśli tak się powszechnie mówi). Jeżeli koniecznie chcemy określić zróżnicowanie elementów elektronicznych pod względem zintegrowania, to zamiast skala integracji należałoby mówić stopień integracji albo lepiej — stopień scalenia (co odpowiada polskiej normie PN-72/T-01600). O elementach elektronicznych mówi się, zresztą, od dawna — układ scalony, a nie — układ zintegrowany.

Podstawowy rodzaj błędów polega na powtórnym nazywaniu tego, co już jest nazwane. Warto więc pamiętać o tym, że komputery istniały już na wiele lat przed pojawieniem się mikroprocesorów i pewne terminy są od dawna ustalone, nie ma więc sensu ich zmieniać. Tak jest, na przykład, z terminem lista rozkazów (ang. instruction set, instruction repertoire). Może nie został on najszcześliwiej dobrany, ale utrwalił się, jest poprawny i dlatego wszelkie inne odpowiedniki polskie, jak np. zbiór instrukcji, zbiór rozkazów, repertuar rozkazów, należy uznać za niepożądane. W nomenclaturze technicznej, jedną z głównych zasad jest zasada jednorodności, stanowiąca wymaganie, aby tylko jeden termin oznaczał jedno pojęcie. Wbrew tej zasadzie postępuje się także wtedy, gdy licznik rozkazów (ang. program counter) nazywa się licznikiem programu. Niekiedy spotyka się też inne nazwy na oznaczenie wskaźnika stosu (ang. stack pointer), jak np. licznik stosu, co również nie jest poprawne. Skoro jeden termin został wybrany i przyjął się, nie należy wprowadzać innego, bo nie ma na to żadnego uzasadnienia.

Powszechna praktyka w tworzeniu terminów z zakresu techniki mikroprocesorowej polega na bezpośredniej adaptacji nazw angielskich, co w wielu przypadkach należy uznać za błąd. Językoznawcy twierdzą bowiem, że zapożyczenia są potrzebne tylko wtedy, gdy nie ma odpowiednich słów polskich. Klasycznym przykładem błędu jest spolszczenie angielskiego wyrazu interface na oznaczenie sprzęgu. Tego rodzaju błędy uznalibyśmy za najpoważniejsze, gdyż szkoda kulturze języka, a odpowiednie wyrazy są najtrudniejsze do wyeliminowania, ponieważ trwale zakorzeniają się w naszej mentalności.

<sup>1)</sup> A. Wiśniewski: Słowniczek mikroprocesorowy polsko-angielsko-francusko-niemiecki. Elementy półprzewodnikowe i układy scalone (Zastosowania. Układy cyfrowe), 10, nr 2(38), s. 53—60, 1982, (wyd. Przemysłowy Instytut Elektroniki)

Nie ma więc prawa obywatelstwa w języku grupa takich terminów, jak: heksadecymalny, decymalny i oktalny. Należy je zastępować wyrazami: szesnastkowy (ang. hexadecimal), dziesiętny (ang. decimal) i ósemkowy (ang. octal). Choć przymiotnika binarny używa się na równi z przymiotnikiem dwójkowy, istnieje silna i uzasadniona tendencja do usunięcia go z języka informatyki. Podobnie, na oznaczenie pojęcia określonego po angielsku nazwą controller nie powinno się używać wyrazu kontroler lecz — sterownik. Nie ma też powodów, aby zestaw rejestrów (ang. register bank) lub moduł pamięci (ang. memory bank) nazywać bankiem rejestrów i bankiem pamięci. Słowa kontroler i bank istnieją w języku polskim lecz nasuwają bardzo odległe skojarzenia, gdyż ich obecne znaczenie daleko odbiega od proponowanego w informatyce. Uwaga ta dotyczy także, choć w znacznie mniejszym stopniu, słowa port (ang. port) — jego proponowany odpowiednik, brama, jest również obciążony zupełnie innym znaczeniem w mowie potocznej. Pamiętajmy też, że w języku polskim nie ma słowa operand — na oznaczenie argumentu operacji (należy mówić argument). Nie powinno się też mówić komenda (ang. command) ani, co gorsze, zlecenie lecz — polecenie. Nie przestrzeganie tej zasady prowadzi do twórców tak karykaturalnych jak „tajmer” (ang. timer), a przecież istnieje bardzo dobry wyraz czasomierz, dokładnie oddający treść odpowiedniego pojęcia.

Myślę, że sprawa jest tak ważna, a jednocześnie nie dość uświadamiamy sobie jej istotę, że warto wesprzeć się głosem autorytetu, którego nie sposób podważyć. Zdaniem prof. Doroszewskiego, „Jeżeli stwierdzimy, że między wyrazem koincydencja a zbieżność zachodzi całkowita tożsamość realnoznaczeniowa, to damy pierwszeństwo zbieżności, jako wyrazowi swojskiemu”. Dochowując wierności tej zasadzie, jestem przekonany, że nie należy mówić kompatybilność lecz zgodność, ponieważ kompatybilność (w informatyce) nie znaczy niczego więcej (ani mniej) niż to, co znaczny zgodność. Nie zawsze jednak można znaleźć dobry odpowiednik polski terminu angielskiego i wtedy trzeba ustąpić przed naporem obcego wyrazu. Tak jest, na przykład, w przypadku terminu strobe. Myślę jednak, że bardziej poprawne i mniej bolesne dla języka byłoby używanie nazwy nieco łagodniejszej niż strob, tj. sygnał strobojący. Okazuje się więc, że właściwe zapożyczenie wyrazu angielskiego nie zawsze jest łatwe i może przybrać formę kalekę. Przykładowo, nie musimy wprowadzać do języka polskiego słowa inicjalizować (ang. initialize), gdyż od dawna istnieje wyraz inicjować (por. Słownik języka polskiego, pod red. M. Szymczaka), który może być użyty w znaczeniu spotykanym w informatyce, tzn. nadawanie wartości początkowej lub — wprowadzenie w stan początkowy.

Inne rodzaje często popełnianych błędów językowych omówimy w następnym numerze INFORMATYKI.

JANUSZ ZALEWSKI

### Od Redakcji

Powyższy odcinek terminologii zawiera nie tylko — zgodnie z tytułem — słownictwo mikrokomputerowe, ale również prezentację poglądów autora na temat zasad tworzenia prawidłowych terminów informatycznych. Wypowiedzi, jakie docierają do naszej redakcji świadczą o tym, że nie wszyscy informatycy zasady te akceptują. W następnym numerze, obok dalszego odcinka terminologii mikrokomputerowej, postaramy się zamieścić wypowiedź prezentującą nieco odmienne stanowisko w kwestiach terminologicznych. Zachęcamy również Czytelników do nadsyłania swych opinii, które są niezbędnym potwierdzeniem społecznego odbioru propozycji terminologicznych.

<p>Rakowski M., Rosiński A. T.: Charakterystyka jednostrukturalnych mikrokomputerów 8-bitowych na przykładzie INTELA 8048</p> <p>INFORMATYKA 1984, nr 6, s. 2</p> <p>Omówienie podstawowych cech obecnie produkowanych na świecie jednostrukturalnych mikrokomputerów 8-bitowych, ilustrowane szczegółowym opisem właściwości funkcjonalnych układu INTEL 8048.</p>	<p>Раковски М., Росиński А. Т.: Характеристика одноструктурных 8-битовых микро-ЭВМ на примере INTEL 8048</p> <p>INFORMATYKA 1984, № 6, стр. 2</p> <p>Обсуждение основных свойств производимых в настоящее время в мире одноструктурных 8-битовых микро-ЭВМ, иллюстрированное подробным описанием функциональных свойств схемы INTEL 8048.</p>
<p>Pawłowski M., Woźniak A.: PROG-2 — uniwersalny programator pamięci stałych</p> <p>INFORMATYKA 1984, nr 6, s. 6</p> <p>Charakterystyka własności i stosowania obecnie produkowanych programowanych pamięci stałych typu PROM oraz szczegółowe omówienie rozwiązań uniwersalnego programatora pamięci stałych PROG-2, opracowanego i uruchomionego w Instytucie Informatyki Politechniki Warszawskiej.</p>	<p>Павловски М., Возняк А.: Универсальный программатор постоянных памяти PROG-2</p> <p>INFORMATYKA 1984, № 6, стр. 6</p> <p>Характеристика свойств и применения производимых в настоящее время программированных постоянных памяти типа PROM и подробное обсуждение решений универсального программатора постоянных памяти PROG-2, разработанного и введенного в эксплуатацию в Отделении вычислительной техники Варшавского политехнического института.</p>
<p>Rzymkowski K.: Mikrokomputerowy system modułarny VME</p> <p>INFORMATYKA 1984, nr 6, s. 9</p> <p>Geneza rozwoju i charakterystyka rozwiązań modułarnego systemu VME. System ten, opracowany na układach firmy MOTOROLA, jest proponowany do uznania go za standard międzynarodowy w zakresie automatyzacji procesów przemysłowych oraz automatyzacji eksperymentów.</p>	<p>Жимковски К.: VME — модульная система Микро-ЭВМ</p> <p>INFORMATYKA 1984, № 6, стр. 9</p> <p>Развитие и характеристика решений модульной системы VME. Система, разработанная на схемах фирмы MOTOROLA, предлагается к признанию в качестве международного стандарта в области автоматизации промышленных процессов и автоматизации экспериментов.</p>
<p>Trechciński R.: MULTIBUS-II — wieloprocesorowy system modułarny</p> <p>INFORMATYKA 1984, nr 6, s. 21</p> <p>Charakterystyka rozwiązań nowego systemu modułarnego MULTIBUS-II, opracowanego na układach firmy INTEL i proponowanego, podobnie jak system VME, do uznania go za standard międzynarodowy w zakresie automatyzacji procesów przemysłowych oraz automatyzacji eksperymentów.</p>	<p>Трехциński Р.: Многопроцессорная модульная система MULTIBUS-II</p> <p>INFORMATYKA 1984, № 6, стр. 21</p> <p>Характеристика решений новой модульной системы MULTIBUS-II, разработанной на схемах фирмы INTEL и подобно тому как VME, предлагаемой к признанию в качестве международного стандарта в области автоматизации промышленных процессов и автоматизации экспериментов.</p>
<p>Rakowski M., Rosiński A. T.: Characteristics of the 8-bit single-chip microcomputers on the INTEL 8048 example</p> <p>INFORMATYKA 1984, No. 6, p. 2</p> <p>Basic properties of the today world-wide manufactured single-chip 8-bit microcomputers, illustrated by detailed description of INTEL 8048 circuit functional characteristics.</p>	<p>Rakowski M., Rosiński A. T.: Eine Charakteristik von 8-Bits-Einschaltkreismikrorechner an Hand von INTEL 8048 Beispiel</p> <p>INFORMATYKA 1984, Nr. 6, S. 2</p> <p>Eine Besprechung von Grundeigenschaften der jetzt produzierten 8-Bits-Einschaltkreismikrorechner, illustriert mit detaillierter Beschreibung der Funktionseigenschaften von INTEL 8048 Schaltkreis.</p>
<p>Pawłowski M., Woźniak A.: PROG-2 — the universal PROM programmer</p> <p>INFORMATYKA 1984, No. 6, p. 6</p> <p>Characteristics of properties and application of the today manufactured PROM storages, as well as detailed discussion of solutions of PROG-2 universal PROM programmer, elaborated in the Data Processing Institute of Warsaw Technical University.</p>	<p>Pawłowski M., Woźniak A.: PROG-2 — ein universeller PROM-Programmierer</p> <p>INFORMATYKA 1984, Nr. 6, S. 6</p> <p>Eine Charakteristik von Eigenschaften und Anwendung der heute produzierten PROM-Speicher und detaillierte Besprechung von Lösungen des PROG-2 universellen PROM-Programmierers, der im Institut für Informatik der Warschauer Technischen Universität erarbeitet und in Betrieb genommen wurde.</p>
<p>Rzymkowski K.: VME — the microcomputer modular system</p> <p>INFORMATYKA 1984, No. 6, p. 9</p> <p>Development and characteristics of the VME modular system solutions. The system, elaborated on MOTOROLA circuits base, is proposed to be an international standard for applications in industrial process and experiments automatization.</p>	<p>Rzymkowski K.: VME — ein modulares Mikrorechnersystem</p> <p>INFORMATYKA 1984, Nr. 6, S. 9</p> <p>Entwicklung und Charakteristik von Lösungen des VME-Modularsystems. Dieses System, erarbeitet auf MOTOROLA-Schaltkreisebasis, wird als internationale Norm im Bereich der Automatisierung industrieller Prozesse und wissenschaftlicher Experimente vorgeschlagen.</p>
<p>Trechciński R.: MULTIBUS-II — the multiprocessor modular system</p> <p>INFORMATYKA 1984, No. 6, p. 21</p> <p>Characteristics of the MULTIBUS-II modular system solutions, elaborated on INTEL circuits base, and proposed, like the VME system, to be an international standard for application in industrial process and experiments automatization.</p>	<p>Trechciński R.: MULTIBUS-II — ein Mehrprozessormodularsystem</p> <p>INFORMATYKA 1984, Nr. 6, S. 21</p> <p>Eine Charakteristik von Lösungen des MULTIBUS-II-Modularsystems, erarbeitet auf INTEL-Schaltkreisebasis, das ähnlich wie das VME-System, als internationale Norm im Bereich der Automatisierung industrieller Prozesse und wissenschaftlicher Experimente vorgeschlagen wird.</p>

■ Japońska firma HITACHI zapowiedziała w sierpniu 1982 przystąpienie do prac nad procesorem szeregowym (ang. array processor) S-810, którego najszybsza wersja ma zapewniać 630 mln operacji zmiennoprzecinkowych na sekundę. Wyprodukowany przez firmę CDC superkomputer CYBER-205 ma szybkość „zaledwie” 400 mln operacji, a komputer Vp-200 japońskiej firmy FUJITSU wykonuje 500 mln operacji zmiennoprzecinkowych w ciągu sekundy. Do roku 1986 HITACHI ma zamiar wyprodukować ok. 90 superkomputerów S-810. Ustalono już nawet ceny ich dzierżawy. Wypożyczenie komputera ma kosztować od 196 tys. dol. miesięcznie za wersję S-810/10 (315 mln operacji) do 275 tys. dol. za wersję S-810/20 — najszybszą. S-810/20 ma mieć do 256 MB pamięci operacyjnej, 32 kanały wejścia-wyjścia o prędkości transmisji 96 MB/s oraz mieć zewnętrzna o pojemności 1 GB (W)

\*

■ Jedną z głównych agencji Departamentu Obrony USA, zajmującą się badaniami podstawowymi, przedstawiła program rozwoju systemów sztucznej inteligencji oraz technologii komputerowej. Planowany koszt badań, bagatelka, 600 mln dolarów. Jeżeli program uda się zrealizować, to sposób, w jaki rozgrywane są współczesne bitwy ulegnie zasadniczym przemianom. Długoterminowa propozycja badań przewiduje konstrukcję nowej generacji komputerów zdolnych widzieć, rozumieć mowę ludzką oraz samodzielnie wyciągać wnioski. Planuje się też stworzenie bezdogowych pojazdów opancerzonych, zdolnych do samodzielnego rekonesansu nieznanego terenu, automatycznych pilotów zdolnych rozumieć głos ludzki, a także rozbudowanego systemu komputerowego wspomagania strategicznego planowania walki. Chcąc ułatwić naprawę nowoczesnej broni, myśli się o zastępowaniu instrukcji obsługi technicznej (w przypadku czołgu M1 ma ona ponad 61 tys. stron) wyrafinowanymi systemami komputerowego diagnozowania uszkodzeń. (T)

\*

■ Rozwój systemów komunikacji optycznej i metod przechowywania danych, takich jak dyski video, stwarza zapotrzebowanie na coraz lepsze mikroelementy optoelektroniczne. W lipcu 1983 dwie największe japońskie firmy elektroniczne HITACHI i MATSUSHITA wraz z przodującym producentem komputerów FUJITSU — ogłosiły przełom w tej dziedzinie. W centralnym laboratorium HITACHI naukowcy usilują osiągnąć coraz większy stopień integracji układów. Na początek postanowili zmieścić całe urządzenie na kostce o wymiarach 1 × 0,6 mm, wykonane z arsenku galu. Urządzenie składa się z lasera półprzewodnikowego, fotodetektora i trzech obwodów (jednego modulującego i dwóch utrzymujących stale natężenie prądu dla lasera i wzmacniacza), zbudowanych z sześciu tranzystorów i dwóch rezystorów. HITACHI zamierza wyprodukować handlową wersję układu, stanowiącą źródło światła o niskim poziomie zakłóceń w systemach dysków optycznych. Fotoelektroniczne urządzenia przyszłości będą zawierać wiele laserów

scalonych na jednej kostce. Jednakże typowe lasery półprzewodnikowe — pracujące wyłącznie przy dużych natężeniach prądu — wytwarzają dużo ciepła. Rozwiązanie proponowane przez FUJITSU to laser multikwantowy, pobierający tylko 20 mA — o połowę mniej niż półprzewodnikowy. Autorzy projektu z laboratorium Kawasaki twierdzą, że niedługo będą mogli zmniejszyć tę wartość dwukrotnie. Kostka składa się z lasera i dwóch tranzystorów polowych, oddzielonych od siebie za pomocą zwykłych metod wytrawiania. Dużym problemem przy wykorzystaniu płaskich kostek optoelektronicznych są „ciemne natężenia” występujące w fotodetektorze, gdy nie przechodzi przez niego światło. Naukowcy z Matsushita Central Research Laboratory skonstruowali kostkę zamieniającą światło na elektryczność, która wykazuje nie więcej „ciemnych natężeń” od normalnego układu. Rezultat ten powinien przyspieszyć szerokie stosowanie układów o wysokiej integracji. (I)

\*

■ Amerykański National Standards Institute ma niemałe kłopoty po opublikowaniu trzy lata temu pierwszej wersji nowego standardu języka COBOL. Okazało się, że wersja języka nie jest kompatybilna z poprzednią, zatem jej przyjęcie równałoby się konieczności ponownego pisania istniejących programów. Również następna próba opisu COBOLU nie spotkała się z życzliwym przyjęciem. Przedstawiciele zainteresowanych firm skądają oświadczenia, że wprawdzie nie zamierzają wytoczyć ANSI procesu, ale są ciągle niezadowoleni z przedstawionych propozycji. Instytut przeprowadził wielką kampanię w środkach masowego przekazu, by wyjaśnić decydem, że standaryzacja języka jest jednak dobrym pomysłem, chociaż problem zgodności kolejnych wersji poważnie ogranicza prowadzone prace. Analiza tysięcy rządowych komputerów przeprowadzona przez National Bureau of Standards wykazała, że przyjęcie najnowszej wersji COBOLU pozwoliłoby Stanom Zjednoczonym zaoszczędzić 72 mln dolarów, dzięki ułatwieniom w pisaniu programów i konserwacji oprogramowania. Zmiana jednego standardu na drugi kosztowałaby rząd ok. 18 mln dolarów. ANSI nadal szuka metod przyspieszenia ostatecznej standaryzacji COBOLU. Postanowiono podzielić język na mniejsze fragmenty i standaryzować je krok po kroku. (I)

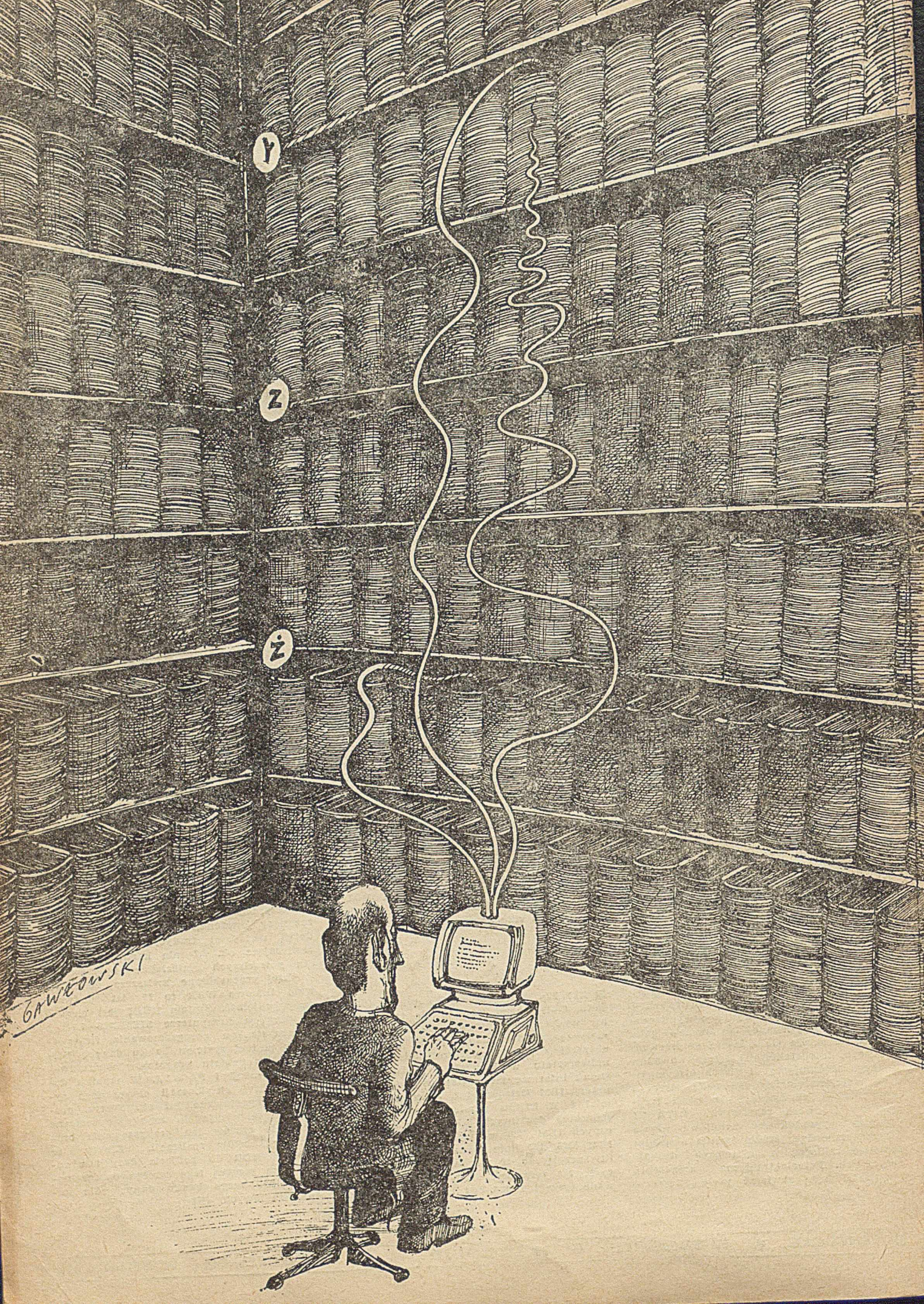
\*

■ XIV Zimowe Igrzyska Olimpijskie (Sarajewo, 8—19 lutego 1984) to nie tylko wielkie wydarzenie sportowe, ale także informatyczne. Nadzór nad całością sprawowali eksperci z IBM i centrum obliczeniowego uniwersytetu w Zagrzebiu. Podstawę systemu informacyjnego stanowiły dwa komputery IBM 4341 model 2, z których każdy miał 8 M bajtów pamięci operacyjnej, 13 jednostek dyskowych 3775,8 taśm magnetycznych IBM 3420 model 8 i trzy drukarki IBM 8203. 102 linie komunikacyjne pozwoliły przyłączyć 60 sterowników nadzorujących 142 terminale 3278, 22 wyspecjalizowane sprzęgi TERA-ZOI i 12 szwajcarskich

systemów pomiaru czasu. Urządzenia te zapewniały stałe połączenia z ośmioma agencjami informacyjnymi w Nowym Jorku, Paryżu, Moskwie, Tokio i Belgradzie. Skonstruowany specjalnie na Igrzyska Olimpijskie, zimowe i letnie, sprzęg TERA-ZOI jest oparty na mikroprocesorze Z80 i zawiera cztery niezależne kanały systemu videotekst. Z punktu widzenia jednostki centralnej, TERA-ZOI emuluje terminal 2780, natomiast na zewnątrz przesyła teksty do ruchomych centrów telewizyjnych. Dzięki temu reporterzy otrzymywali natychmiast listy uczestników, informacje o nich i analizy uzyskanych wyników. Wszystkie informacje przedstawiono w siedmiu centrach prasowych w miejscach rozgrywania konkurencji oraz w głównym centrum Alipasin Most, w którym znajdowało się 25 terminali wraz z operatorkami ułatwiającymi ich obsługę. Oprócz tego dziennikarze, goście i organizatorzy korzystali z 12 dodatkowych punktów informacyjnych. System pracował 24 godziny na dobę od 30 stycznia do 20 lutego i służył nie tylko celom informacyjnym, ale także prowadził sprawy finansowe, administracyjne, a nawet — planistyczne. Bazy danych obejmowały sportowców, działaczy, dziennikarzy, fotoreporterów, techników telewizyjnych — w sumie ok. 13 tys. osób. Opracowanie oprogramowania koniecznego do funkcjonowania systemu wymagało współpracy dwudziestu ekspertów, zaś w trakcie samych igrzysk zatrudniono przy nim 450 osób, w tym 250 studentów kierunków informatycznych i filologicznych. (M)

\*

■ Pierwszy w obszarze krajów języka niemieckiego bank danych telewizji uruchomiła znana sieć radiowo-telewizyjna Westdeutsche Rundfunk (WDR). Bank ten funkcjonuje w oparciu o informatyczny system wyszukiwania informacji RUDI (Rundfunk-Dokumentations- und Informationssystem) i w chwili obecnej obejmuje ewidencję ok. 100 tys. kaset zawierających łącznie ok. 50 mln m filmów. Zapis informacyjny każdej kasety zawiera jej dane formalne oraz zwięzłą charakterystykę treści umieszczonego w nim filmu (nagrania audycji telewizyjnej), pozwalającą szybko odszukać materiał potrzebny do przygotowania nowej audycji lub widowiska telewizyjnego. Ze względu na olbrzymie rozmiary ewidencji filmów, prowadzonej dotąd metodami tradycyjnymi (kartoteka zawierająca ok. 2,7 mln kart), poszukiwania potrzebnych materiałów były coraz bardziej pracochłonne, zwłaszcza, że archiwum filmowe WDR co 10 lat podwaja swoją objętość. System RUDI jest eksploatowany na komputerze SIEMENS 7.541 z wykorzystaniem oprogramowania firmowego w postaci systemu banku danych GOLEM oraz pakietu programów wyszukiwania informacji i PASSAT. W wyniku dotychczasowej eksploatacji omawianego systemu przygotowano już ok. 80 tys. programów telewizyjnych, uzyskując znaczne skrócenie czasów ich przygotowania oraz obniżkę kosztów realizacji. Efekty te skłoniły WDR do podjęcia decyzji utworzenia za pomocą tego samego oprogramowania podobnego banku danych dla audycji radiowych. (K)



Y

Z

Z

GAWŁOWSKI