



P.1877 | 90

7

1990

informatyka

**System baz danych CX-DMOS
Sieci MAP i TOP
Edytor schematów elektronicznych**

KOLEGIUM REDAKCYJNE:

mgr Jarosław DEMINET,
dr inż. Wacław ISZKOWSKI,
mgr Teresa JABŁONSKA,
(sekretarz redakcji)
Władysław KLEPACZ
(redaktor naczelny)
dr inż. Wojciech MOKRZYCKI,
mgr inż. Jan RYZKO,
mgr Hanna WŁODARSKA.

PRZEWODNICZĄCY RADY PROGRAMOWEJ:

Prof. dr hab.
Juliusz Lech KULIKOWSKI

WYDAWCA: Wydawnictwo Czasopism i Książek Technicznych, SIGMA NOT

Spółka z o.o.
ul. Biała 4,
00-950 WARSZAWA
skrytka pocztowa 1004

Redakcja: 01-552 Warszawa,
Pl. Inwalidów 10 p. 128, 136
tel. 39-14-34

Materialów nie zamówionych
redakcja nie zwraca

DRUK:
PRASOWE ZAKŁADY GRAFICZNE
RSW „PRASA-KSIĄZKA-RUCH”
ul. Dworcowa 13,
85-950 BYDGOSZCZ
zam. 981/90
Obj. 4,0 ark. druk. Nakład 5250 egz.

Cena egzemplarza 5800 zł

**W sprawach ogłoszeń
prosimy zwracać się bez-
pośrednio do Redakcji**

W numerze:

	Strona
System baz danych CX-DMOS (I) <i>Lubomir Jurczak, Wojciech Warski</i>	1
GT-HOLMES graficzna baza danych <i>Henryk Rybiński, Piotr Sobolewski</i>	4
Sieci MAP i TOP. Przegląd standardów i przykład aplikacji <i>Andrzej Białas, Aleksandra Bojduk</i>	9
Konwersja danych między pakietami micro CDS/ISIS i dBase III Plus <i>Dorota Szczawińska, Bogdan Trawiński</i>	13
Edytor schematów elektronicznych <i>Andrzej Maciej Wierzb</i>	17
Sprzętowe wzmocnienie obrazów cyfrowych w czasie rzeczywistym <i>Nguyen Kim Sach</i>	20
FidoNet międzynarodowa amatorska sieć komputerowa <i>Jan Stózek</i>	22
Listy	24
Ulepszona wersja programu porządkującego według polskiego alfabetu	
Z kraju	25
INFOSYSTEM '90	
Ze świata	26
Współczesne pamięci masowe	
Rynek pamięci operacyjnych	

zBITki

III okł.

W najbliższych numerach:

- Witold Rekuć charakteryzuje programowe rozwiązania systemu informatycznego prowadzące do zmniejszenia jego wrażliwości na zmiany w strukturze danych.
- Miłosz Muszyński opisuje fizyczne struktury systemów baz danych.
- Nguyen Kim Sach omawia wzmocnienie obrazów telewizyjnych metodami komputerowego przetwarzania obrazów.
- Witold R. Rudolf prezentuje zastosowanie komputerów do wspomagania nauczania, na przykładzie programu edukacyjnego JU-LEK.
- Jan Ryżko i Zdzisław Wrzeszcz podają różne rozwiązania techniczne optycznej pamięci dyskowej i omawiają sytuację krajową w tej dziedzinie.
- Mariusz Postół przedstawia wyniki analizy zagadnień związanych z konstrukcją niezawodnego oprogramowania systemowego dla przemysłowych systemów czasu rzeczywistego.

WARUNKI PRENUMERATY

Prenumeratory zbiorowi – jednostki gospodarki społecznej, instytucje i organizacje społeczne zamawiają prenumeratę dokonując wpłat wyłącznie na blankiecie „wpłata zamówienie” (jest to „polecenie przelewu” rozszerzone dla potrzeb Wydawnictwa o część dotyczącą zamówienia). Blankiety te będą dostarczane dotychczasowym prenumeratom przez Zakład Kolportażu. Nowi prenumeratory otrzymują je po zgłoszeniu zapotrzebowania (pisemnie lub telefonicznie) w Zakładzie Kolportażu, w Radach Wojewódzkich NOT bądź w Redakcjach czasopism.

Prenumeratory indywidualni – osoby fizyczne zamawiają prenumeratę dokonując wpłaty w UPT lub NBP na blankiecie NBP. Na odwrocie wszystkich odcinków blankietu należy wpisać tytuł czasopisma, okres prenumeraty, liczbę zamawianych egzemplarzy oraz wartość wpłaty. Wpłacać należy na konto: Państwowy Bank Kredytowy III/O Warszawa nr 370015-7490-139-11.

Prenumerata ulgowa – przysługuje wyłącznie osobom fizycznym – członkom SNT, studentom i uczniom szkół zawodowych. Warunkiem prenumeraty ulgowej jest poświadczenie blankietu wpłaty (przed jej dokonaniem) na wszystkich odcinkach pieczęcią Koła SNT, wyższej uczelni lub szkoły. Sposób zamawiania prenumeraty ulgowej jest taki sam jak prenumeraty indywidualnej. W prenumeracie ulgowej można zamówić tylko po jednym egzemplarzu każdego czasopisma.

Uwaga! Miesięcznik „Aura” może być zamawiany w prenumeracie ulgowej również przez uczniów szkół ogólnokształcących. **Prenumerata ze zleceniem wysyłki za granicę** – zamawia się tak, jak prenumeratę indywidualną. Dodatkowo należy podać na blankiecie wpłaty nazwisko i dokładny adres odbiorcy.

Cena prenumeraty ze zleceniem wysyłki za granicę jest dwukrotnie wyższa.

Wpłaty na prenumeratę przyjmowane są w terminach:

- do 10 listopada na każdy kwartał, I i II półrocze oraz cały rok następnym.
- do 28 lutego na II, III i IV kwartał oraz II półrocze.
- do 31 maja na III i IV kwartał oraz II półrocze.
- do 31 sierpnia na IV kwartał.

Zmiany w prenumeracie można zgłaszać pisemnie tylko w wyżej wymienionych terminach.

Informacji o prenumeracie udziela – Zakład Kolportażu Wydawnictwa NOT SIGMA (ul. Bartycka 20, 00-716 Warszawa), skr. poczt. 1004, 00-950 Warszawa, tel. 40-00-21 wew. 248, 249, 293, 297, 299 lub 40-30-86 i 40-35-89.

Egzemplarze archiwalne czasopism – można nabyć za gotówkę w Klubie Prasy Technicznej, Warszawa ul. Mazowiecka 12 (tel. 26-80-16), lub zamówić pisemnie. Zamówienia na egzemplarze archiwalne czasopism przyjmuje: Zakład Kolportażu, Dział Handlowy, 00-950 Warszawa, skr. poczt. 1004 (tel. 40-37-31), na rachunek dla instytucji lub za zaliczeniem pocztowym dla osób fizycznych.

Cena egzemplarza: normalna 5800 zł, ulgowa 1160 zł

Uwaga! Podane ceny mają charakter wstępny i mogą ulec zmianie, w związku z powyższym Wydawnictwo zastrzega sobie prawo żądania dopłat do już opłaconej prenumeraty.

P.1877



**LUBOMIR JURCZAK
WOJCIECH WARSKI**
Przedsiębiorstwo Zagraniczne COMPUTEX
Warszawa

System baz danych CX-DMOS (1)

Wielodostępny system baz danych CX-DMOS to produkt o stosunkowo długiej historii, jeśli za skalę czasu przyjąć rozwój zastosowań mikrokomputerów w Polsce. Trzy lata pracy owocują obecnie systemem, który ma już za sobą choroby wieku dziecięcego i stwarza interesujące możliwości realizacyjne. O tych właśnie możliwościach systemu oraz zrealizowanych pomysłach traktuje poniższy artykuł.

CX-DMOS jest produktem zbyt dużym, by włączyć go w ramy określenia „system operacyjny”. Jest to raczej realizacja pewnej koncepcji wielostanowiskowej pracy z bazami danych na mikrosprzęcie. Dyskusje o tym, czy komputery mające w nazwie przymiotnik „osobiste” są najlepszym wyjściem przy realizacji systemów baz danych, powoli zaczynają zanikać, głównie dzięki istotnemu zwiększeniu się mocy obliczeniowej tej kategorii komputerów oraz rozwojowi metod zabezpieczenia danych.

W tej sytuacji brak odpowiedniego oprogramowania powoduje, że komputery osobiste są rzadko stosowane do realizacji poważniejszych zadań, tzn. takich, których nie można efektywnie zrealizować np. korzystając z dBase. Stworzeniem takiego oprogramowania mało kto jest zainteresowany na Zachodzie – od czegoż bowiem mają od lat dobre minikomputery i duże maszyny. W Polsce sytuacja długo jeszcze będzie krańcowo odmienna: prężny i konkurencyjny ekonomicznie rynek mikrokomputerów wręcz zmusza do powszechnego ich stosowania. Cena, łatwość instalacji, duże już moce obliczeniowe, a nawet całkiem przyzwoita niezawodność markowego sprzętu, są istotnymi argumentami „za” w rozważaniach na temat możliwych dróg komputeryzacji przedsiębiorstwa, urzędu czy instytutu.

Obecna wersja systemu CX-DMOS została zrealizowana na podstawie systemu plików DOSa. Jak wiadomo, DOS nie jest systemem wielodostępnym, toteż zarządzaniem rozdzielnymi zadaniami zajmuje się specjalnie w tym celu napisane jądro. Wersja CX-DMOS, planowana dla systemu operacyjnego OS/2, będzie już wykorzystywać w warstwie zarządzania i komunikacji między zadaniami cechy samego systemu.

DOS wymaganych mechanizmów niestety nie dostarczał, stąd konieczność jego „uzupełnienia”, które de facto sprowadziło go wyłącznie do roli modułu wejścia-wyjścia dla dysku twardego.

Zanim CX-DMOS zostanie bardziej szczegółowo przedstawiony, kilka słów o formach jego organizacji i dystrybucyjnego punktu widzenia oraz wymaganiach sprzętowych.

Stwierdzenie podstawowe: CX-DMOS jest kompletnym środowiskiem programowania. Oznacza to, że dostarczany zestaw narzędzi całkowicie zaspokaja potrzeby programistów aplikacyjnych oraz administratora systemu. W skład zestawu wchodzi bowiem pracujące pod kontrolą systemu DOS programy, służące bezpośrednio tworzeniu nowych programów użytkowych: kompilator języka CX-BASE, symboliczny program uruchomieniowy (debugger), programy definiowania ekranów, wydruków i struktur baz danych oraz wiele innych. Programom tym towarzyszy pokaźna, czterystustronicowa dokumentacja, niezależnie od „podręcznej” bazy danych, zorganizowanej przy użyciu programu „Norton Guides”.

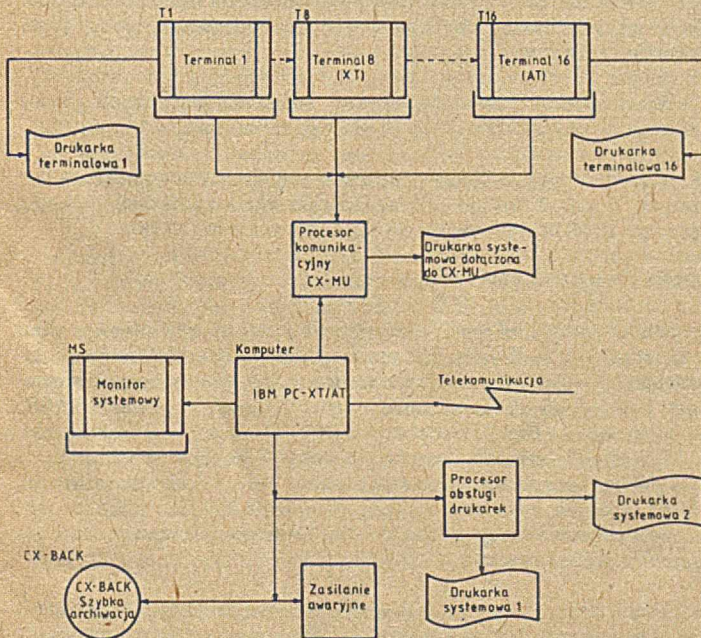
Taka postać zestawu dystrybucyjnego wykracza oczywiście daleko poza potrzeby użytkownika, eksploatującego jedynie zamknięty zestaw programów użytkowych. Dla tych ostatnich system CX-DMOS jest więc oferowany jedynie w postaci systemu operacyjnego, towarzyszącego konkretnemu programowi użytkowemu, oraz narzędzi konfiguracyjnych.

W obu przypadkach instalowanie środowiska CX-DMOS – pełnego bądź wykonawczego – jest maksymalnie uproszczone dzięki interakcyjnemu programowi instalacyjnemu. Program ten w sposób inteligentny (odporny na błędy użytkownika) buduje na dysku całą strukturę podkatalogów, wypełnioną plikami pochodzącymi z kolejnych dyskietek instalacyjnych.

Przedstawiona na rys. 1 konfiguracja sprzętowa jest klasyczna: kilka bądź kilkanaście terminali klasy VT52 (które doczekały się już zainteresowania co najmniej kilku producentów krajowych), dołączonych przez łącza szeregowo RS 232 (bezpośrednio, przez modemy lub pętlę prądową) do komputera centralnego klasy IBM PC/XT lub (lepiej) AT. Opis będzie bardziej zrozumiały, gdy zdefiniuje się dwa problemy, których rozwiązanie bezpośrednio rzutuje na możliwości systemu wielodostępnego.

Problem pierwszy to komunikacja z terminalami. Jak bardzo może ona obciążyć procesor centralny, wiedzą najlepiej użytkownicy Xenixa, którzy z reguły nawet nie próbują dołączać więcej niż 2-4 terminale. Rozwiązaniem, które zastosowano w systemie CX-DMOS jest wielokanałowa karta wejść-wyjść szeregowych CX-MU z własną pamięcią i procesorem – czyli w istocie osobny, dedykowany komputer komunikacyjny. Jego zadaniem jest nie tylko przyjmowanie i wysyłanie pojedynczych znaków, ale też formalna kontrola danych oraz niezbędne przekodowanie. Procesor centralny otrzymuje więc z terminala gotowe, duże porcje danych, które przetwarza bez jakichkolwiek zatrzymań. Karta CX-MU jest obecnie dostępna w wersji 8-kanałowej, natomiast wkrótce będzie dostępna wersja 16-kanałowa z procesorem 80186.

Problem drugi to dostępna pamięć operacyjna. Pomieszczenie niezależnych programów z kilku terminali jednocześnie wymaga więcej niż 640 KB pamięci obsługiwanej przez DOS. Ponieważ jednak obecna wersja systemu pracuje wyłącznie w trybie zwykłym procesorów rodziny Intel, nie ma więc dostępu do pamięci rozszerzonej (ang. *extended memory*) i widocznej dla procesorów 80286/386 w trybie pracy chronionej. Zastosowano tu proste, a bardzo skuteczne, rozwiązanie w postaci dodatkowej karty pamięci EMS (ang. *expanded – stronicowana*), które



Rys. 1. Konfiguracja dedykowanego systemu baz danych CX-DMOS

nie tylko usuwa ograniczenia w pojemności pamięci, ale również skutecznie chroni poszczególne zadania systemu przed niepożądanymi skutkami ewentualnych błędów. Każde z zadań z założenia operuje tylko na swojej pamięci „prywatnej” (i oczywiście jądra systemu, które jest wielodostępne). Z ich punktu widzenia jest więc całkowicie wystarczające, gdy w chwili przełączania kontekstu procesora są udostępniane w oknie (64 KB) właściwe dla określonego zadania strony pamięci EMS.

Wykorzystanie pamięci EMS jako pamięci operacyjnej zadań terminalowych wprowadza pewną nadmiarowość do pracy systemu, ponieważ czas każdorazowego przełączania kontekstu jest znacznie wydłużony. Jest to jednak rozsądna cena za zyski płynące z jej użycia.

SYSTEM OPERACYJNY

Przyjrzyjmy się teraz bliżej konstrukcji systemu operacyjnego.

Zadania

Zdefiniowane są trzy klasy zadań, rozumianych jako niezależne ciągi wykonawcze rozkazów, którym przydzielane są kolejne odcinki czasu procesora (zadania działają na niezależnych danych i stosach, nie mając możliwości komunikowania się na bieżąco, ponieważ potrzeba taka nie wynika z zakresu zastosowań).

W klasie **zadań systemowych** funkcjonują: spooler wydruków, unspoolery drukarek, zadanie operatora konsoli głównej oraz debuggera. Przewiduje się rychłe wprowadzenie zadania telekomunikacyjnego, które będzie obsługiwać transmisję danych przez modem. Operator konsoli głównej decyduje o szczególnych akcjach systemu, takich jak jego zamknięcie, uruchomienie śledzenia, archiwowanie danych bądź „ręczne” uaktywnienie zadania.

Drukarki zostały potraktowane w projekcie systemu ze szczególną uwagą (rys. 1), ponieważ większość zastosowań baz danych wymaga emisji obszernych zestawień okresowych i dokumentów. Ważne więc było, by użytkownik otrzymał do dyspozycji adekwatne mechanizmy wykorzystania drukarek, które zostały podzielone na dwie grupy, związane z ich fizyczną lokalizacją i przeznaczeniem.

Drukarki systemowe (do 4), dołączone bezpośrednio do komputera, są dostępne ze wszystkich terminali (czyli dla wszystkich programów użytkowych). Wywołując z konsoli głównej zadanie spoolera można również „ręcznie” kierować dowolne pliki do wydruku, a także operować plikami znajdującymi się już w centralnej kolejce wydruków (zmieniać żądany typ drukarki, wstrzymywać drukowanie, przeszeregowywać wydruk, ponawiać wydruki itp.).

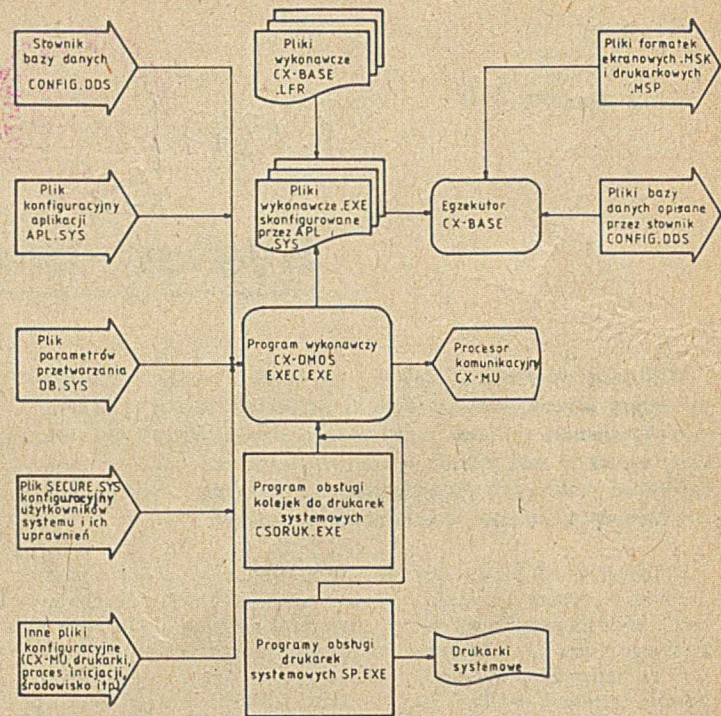
Odmienne potraktowano drukarki terminalowe – są one całkowicie autonomiczne, niewidoczne dla systemu. Sposób fizycznego dołączenia określa ich lokalizację w rejonie konkretnego terminala.

Zadania kontrolne są okresowo uaktywniane mechanizmem zegarowym w celu autotestowania systemu. Mogą też one podejmować akcje ratunkowe w razie poważnych awarii komputera (np. awaria procesora perferencyjnego CX-MU, błąd sumy kontrolnej pamięci).

Podstawową pracę użytkową wykonują zadania z klasy **terminalowej**. Każde z nich obsługuje polecenia konkretnego terminala, przekazywane do (z) procesora CX-MU w postaci formatek ekranowych. Koncepcja formatki – zapożyczona z dużych komputerów – pozwala przetrzymać kontrolę formalną wprowadzanych danych do procesora CX-MU. Procesor główny może więc równolegle obsługiwać gotowe, duże bloki danych, otrzymywane z poszczególnych terminali.

KONFIGUROWANIE SYSTEMU

Dla systemu tej klasy co CX-DMOS, naturalny jest wymóg wielopłaszczyznowej konfiguralności. Tak więc zadeklarowania wymagają poszczególne zestawy programów, dostępne dla użytkowników, oraz sam system operacyjny. Konfigurowaniu programów służy cały zestaw narzędzi, który zostanie omówiony w drugiej części artykułu poświęconej programowaniu w systemie CX-DMOS. Poniżej przedstawiono konfigurowanie samego systemu w odniesieniu do schematu przetwarzania danych w systemie CX-DMOS (rys. 2).



Rys. 2. Przetwarzanie danych w systemie CX-DMOS

Pliki konfiguracyjne są zgrupowane w katalogu \SYS. Kluczowym plikiem w tym katalogu jest *DB.SYS*, przygotowywany programem *DBSET.EXE*. Z zawartości pliku wynikają wielkości wszystkich buforów systemu, liczba aktywnych terminali i ograniczenia liczby plików, ścieżki dostępu do plików danych oraz typy i sposób dołączenia drukarek systemowych. Definicje te są konieczne do właściwego rozplanowania systemu w pamięci, zanim jeszcze rozpocznie on normalną pracę.

Inne pliki konfiguracyjne to:

CONFIG.DDS – plik generowany programem *RGEN.EXE*, zawierający opis rekordów użytych baz danych,
APL.SYS – generowany programem *APLSET.EXE* plik konfiguracyjny aplikacji, opisujący każdy z programów dopuszczonych do pracy w danej instalacji systemu,
SECURE.SYS – generowany programem *SDEFINE.EXE* plik definiujący użytkowników systemu; rekordy tego pliku są dostępne programom użytkowym,
CXMUINI.SYS – generowany programem *CXMUINI.COM* plik definiujący cechy fizyczne dołączonych terminali (sekwencje sterujące, parametry transmisji).

Wynikowy, w pełni rozwinięty na podstawie powyższych plików konfiguracyjnych, obraz pamięci przedstawiono na rys. 3.

Dla instalacji systemu, zawierającej więcej niż trzy terminale, „obszar zadań” z rys. 3 zostaje przeniesiony do okna niezbędnej wówczas pamięci EMS, a więc zwykle do segmentu o adresie *D000H*.

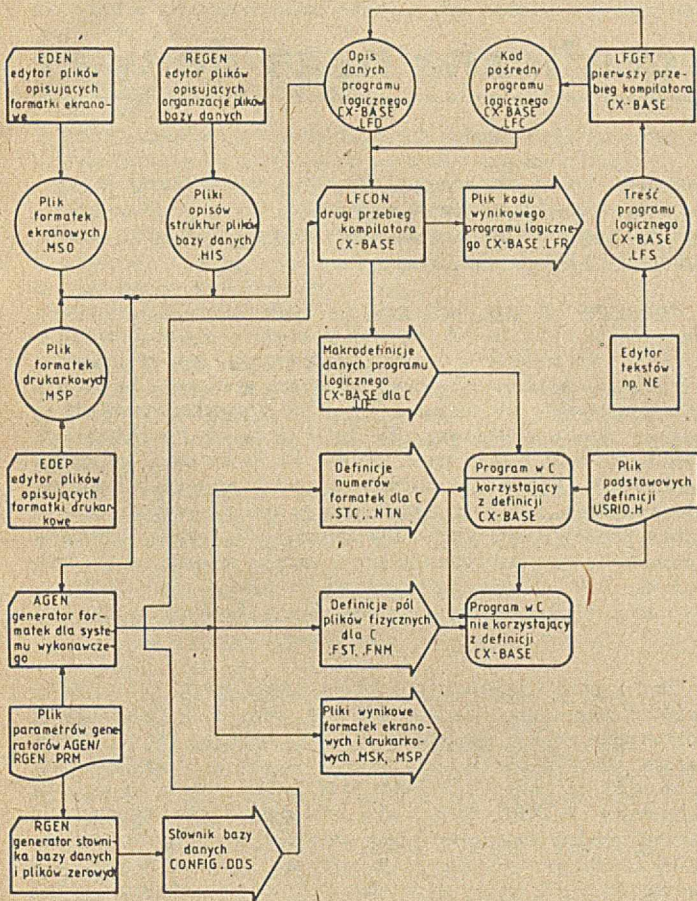
BUDOWA SYSTEMU

Moduł opisany na rys. 3 jako jądro CX-DMOS jest złożeniem kilku elementarnych podsystemów – między innymi interpretera kodu pośredniego języka CX-BASE, arbitra transakcji, programu archiwowania. Udostępnia on zespół funkcji wywoływanych przerwaniem *99H*, analogicznie do *INT 21H* w systemie DOS. Funkcje te – w liczbie blisko 50 – wykonują polecenia dotyczące globalnych zasobów systemu, takich jak pamięć operacyjna, kolejka wydruków, bufory drukarek, głównego terminala, plik raportu dziennego, procesor CX-MU i kilka innych. Z natury rzeczy funkcje są wielodostępne, a obszary krytyczne osłonięte semaforami systemowymi.

Moduł zarządzający szeregowaniem zadań (*dispatcher & scheduler*) jest odpowiedzialny za organizację wykorzystania procesora przez aktywne zadania. Centralny zasób modułu – kolejka dostępu do

procesora – jest typu cyklicznego (prostego), bezpriorytetowego, z możliwością umieszczenia zadania w procesorze w trybie awaryjnym, tzn. przy najbliższym przerwaniu zegara. Zadanie nie może znajdować się tylko w jednym z trzech stanów: wykonywane, oczekujące na procesor, zawieszona. Tak prosta organizacja owocuje efektywnym i łatwym do śledzenia kodem.

W połączeniu z zespołem odpowiednich instrukcji języka CX-Base, który zostanie przedstawiony w drugiej części artykułu, pliki listowe są bardzo efektywnym (w sensie czasu dostępu i oszczędności miejsca) narzędziem dostępu do danych zapamiętanych w postaci dużej liczby krótkich rekordów. Przykładowo, w systemie księgowości, zbudowanym na bazie CX-DMOSA, w plikach stronicowo-listowych są przechowywane tablice wskaźników do innych plików bazy księgowej.



Rys. 3. Przepływ informacji w procesie tworzenia aplikacji w języku C pod nadzorem systemu CX-DMOS

Opisywany moduł udostępnia również – przez przerwanie 9CH – zestaw funkcji, pozwalających indywidualnie sterować wykorzystaniem czasu procesora przez dane zadanie. Przykładowo, jako zasadę przyjęto, że w systemie nie jest realizowane jakiegokolwiek aktywne (cyklicznie badające określone znaczniki) oczekiwanie. W zamian jest wywoływana funkcja zwalniająca natychmiast procesor dla innego zadania, a zadanie zwalniane jest umieszczone na końcu kolejki. Zadanie może też dobrowolnie zawiesić się w oczekiwaniu na określone wykonanie polecenia. Ten model działania zaimplementowano np. we współpracy z procesorem CX-MU.

Ostatnim z wyróżnionych na rys. 3 modułów jest ISAM. Jest to zespół procedur implementujących indeksowo-sekwencyjny, listowo-stronicowany i swobodny dostęp do plików. O ile algorytm zrównoważonego drzewa binarnego w dostępie do plików indeksowo-sekwencyjnych i swobodnych (w których numer rekordu jest indeksem) jest dość powszechnie znany, o tyle organizacja listowa plików wymaga objaśnienia.

W implementacji tej organizacji wykorzystuje się – przy dostępie do poszczególnych stron – mechanizmy modułu ISAM, można więc powiedzieć, że stanowi ona specyficzne wykorzystanie tego modułu. Każda strona ma nagłówek zawierający wskaźniki dwukierunkowej listy stron oraz opis pomocniczy danych zawartych na stronie.

Wskaźniki listy są automatycznie modyfikowane w razie podziału strony przy zapisie danych. Wolna przestrzeń 512-bajtowych str... :t przeznaczona na przechowywanie sekwencyjnie ułożonych rekordów.

Konferencje

Zintegrowane, inteligentne systemy informatyczne

W dniach od 23 do 27 września br. odbędzie się w zamku w Tucznie k. Piły (Dom Pracy Twórczej SARP) międzynarodowa konferencja pod nazwą „Zintegrowane, inteligentne systemy informatyczne”. Organizatorami konferencji są Uniwersytety w Dortmundzie i w Oldenburgu (RFN) oraz Akademia Ekonomiczna w Poznaniu.

Program konferencji obejmuje 22 referaty autorów polskich, niemieckich i szwajcarskich, wygłaszane w ramach siedmiu sesji tematycznych (systemy: eksperckie, tekstowe, prezentacji wiedzy, informacyjne dla biur, wyszukiwanie informacji, systemy narzędziowe i aplikacyjne, technologia baz danych). Oprócz wymienionych sesji program przewiduje dyskusję panelową na temat „Informatyka w nowych i starych demokracjach – szanse i zagrożenia”, a także całodzienną wycieczkę do Poznania, ukierunkowaną na zainteresowania gości zagranicznych. Wycieczka ta obejmie m. in. zwiedzanie przedsiębiorstw informatycznych oraz Akademii Ekonomicznej. Językami konferencji będą niemiecki oraz angielski.

Informacje na temat warunków uczestnictwa oraz zgłoszenia udziału w konferencji: dr inż. Witold Abramowicz, Akademia Ekonomiczna w Poznaniu, 61-720 Poznań, Al. Niepodległości 10, tel. 602-51

KARTY PRZETWORNIKÓW ANALOGOWO-CYFROWYCH do komputerów IBM PC/XT/AT technologia „CONVERT”; podzespoły ANALOG DEVICES

- do 32 wejść analogowych
 - napięcia wejściowe: $\pm 5V$, $\pm 10V$, 0-10V, 0-20V
 - dynamika 12 bitów
 - przetwornica DC/DC zasilająca część analogową
 - precyzyjny wzmacniacz pamiętająco-próbkujący
 - dwa analogowe kanały wyjściowe
 - napięcia wyjściowe: $\pm 5V$, $\pm 10V$, 0-10 V, 0-20V
 - trzy 16-bitowe progromowalne liczniki
 - układ generacji przerwań
 - współpraca z DMA (próbkowanie 'w tle')
 - licznik zdarzeń
 - osiem wejściowych
- oraz osiem wyjściowych kanałów cyfrowych

**DO TEGO NIEODPŁATNIE:
OPROGRAMOWANIE PODSTAWOWE
oraz POMOC W INSTALACJI**

INFORMACJE:

'CONVERT T.S.'

53-143 Wrocław, ul. Orla 2A, Tel. 61-92-28

0/4/90

GT-HOLMES – graficzna baza danych

Na początku lat osiemdziesiątych pojawiły się pierwsze publikacje dotyczące graficznych baz danych [1]. W swoim rozwoju bazy graficzne korzystają z korzeni jakimi są klasyczne bazy danych. Biorąc za punkt wyjścia klasyczne bazy danych, można wyróżnić trzy zasadnicze płaszczyzny w których występują trudności podczas projektowania i realizacji baz graficznych:

- sprzęż z użytkownikiem, sposób dostępu do danych graficznych i manipulacja nimi,
- model danych użyty do realizacji i oparty na nim język,
- reprezentacja danych przestrzennych i sposób ich indeksowania (dane przestrzenne są wielowymiarowe w przeciwieństwie do danych alfanumerycznych, które są jednowymiarowe).

W artykule opisano projekt systemu graficznej bazy danych GT-HOLMES. Oparto go na dedukcyjnej bazie danych HOLMES [4]. Dedukcyjna strona systemu nie różni się od wzorca, jakim jest system HOLMES opisany w [4, 8]. Dlatego uwagę skupiono na warstwie sprzętu z użytkownikiem i języku zapytań w warstwie mechanizmów obsługi danych graficznych.

Większość dotychczasowych systemów graficznych stanowi specjalizowane oprogramowanie, które tworzone na specjalne potrzeby i z myślą o specjalnym sprzęcie. Coraz częściej pojawiają się przy tym problemy związane z gromadzeniem dużych ilości informacji graficznych oraz ich opisu. Przykładowo, użytkownik systemu CAD operuje wieloma oddzielnymi rysunkami, przechowując atrybuty obiektów w standardowym systemie obsługi baz danych. Informacje alfanumeryczne są jedynie biernym opisem informacji graficznej. Powiązanie między tymi systemami zapewnia specjalny sprzęg zarządzany bezpośrednio przez użytkownika.

Graficzna baza danych jest to system, w którym zarządza się zarówno danymi alfanumerycznymi, jak i informacją graficzną. System taki powinien mieć jeden spójny język opisu i operowania danymi, umożliwiający użytkownikowi równoprawne operowanie danymi graficznymi i alfanumerycznymi. W graficznych bazach danych, w trakcie analizy wyrażen języka generuje się oddzielnie polecenia dla procesora danych alfanumerycznych i procesora danych przestrzennych (rys. 1) [10].

Z punktu widzenia użytkownika możliwości operowania danymi graficznymi muszą być równie proste, jak w klasycznych systemach obsługi baz danych. Systemy te umożliwiają przeglądanie i redagowanie

danych w postaci tabelarycznej. Bazy graficzne natomiast powinny ponadto zapewniać możliwość wyświetlania obrazów, zmiany skali, a przede wszystkim korygowania obiektów graficznych (ich kształtu, położenia itp.).

Historycznie, baza graficzna najczęściej była postrzegana jako typowa baza relacyjna [1, 6, 10]. Nie brak też poglądu, iż model relacyjny, mimo swojej prostoty i łatwości implementacji, nie przystaje do reprezentowania skomplikowanej informacji graficznej. Informacja opisująca obraz graficzny ma cechy, które elegancko wyraża się za pomocą modelu obiektowego [9]. Takie cechy jak hierarchiczność, korpuskularność, dziedziczenie, idealnie pasują do opisu informacji graficznej. Realizacja tych zależności w modelu relacyjnym jest oczywiście możliwa, ale wymaga zdefiniowania relacji powiązań, co przy pewnej liczbie poziomów powoduje zatracenie klarowności schematu. Model obiektowej bazy danych, mimo wielu ciekawych i pożądanych cech, nie doczekał się jeszcze jednolitego, formalnego opisu. Dodatkowo traci się w nim prostotę modelu relacyjnego. Dyskusję na ten temat i próbę powiązania obu modeli przeprowadzono w [11, 12].

Oprócz prób z modelem relacyjnym i obiektowym pojawiają się próby zastosowania innych modeli baz danych. Najczęściej są to rozszerzenia modelu relacyjnego. Charakterystyczny jest tu projekt systemu POSTGRES [13], który miał spełniać wymagania zaawansowanych aplikacji typu CAD/CAM. POSTGRES oparto na systemie relacyjnym INGRES, dlatego ideą było ograniczenie do minimum zmian w modelu relacyjnym. Innym podejściem jest utworzenie nadbudowy obiektowej nad istniejącym systemem; przykładem może być system R^2D^2 [7] oparty na modelu NF^2 [3]. Przykładem systemu opartego na modelu E-R jest system opisany w [5].

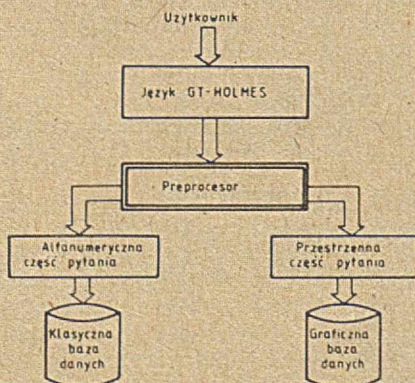
Z względu na bogatą strukturę opisu informacji graficznej pożądane wydaje się zastosowanie dedukcji jako narzędzia do operowania tą strukturą. Uzyskuje się w ten sposób możliwość wnioskowania z faktów zawartych w bazie przy wykorzystaniu uniwersalnych zależności zdefiniowanych w schemacie. Przykładem systemu o cechach dedukcyjnych może być system łączący elementy obiektowości z logiką, opisany w [9]. Innym ciekawym przykładem jest system IIDS, w którym oprócz logiki zastosowano metodę reprezentacji danych przestrzennych (ang. 2-D strings) wykorzystywaną bezpośrednio w procesie wnioskowania [2].

System opisany w artykule łączy w sobie cechy graficznej i dedukcyjnej bazy danych. W proponowanym systemie język dostępu oparto na logice, otrzymując w ten sposób silny aparat do formułowania zapytań, definicji i zasad spójności, wspólny dla danych graficznych i alfanumerycznych.

FUNKCJE GRAFICZNEJ BAZY DANYCH

Z funkcjonalnego punktu widzenia można określić pewne – minimalne – wymagania nakładane na system graficznej bazy danych:

- wszelkie dane graficzne powinny być wyświetlane na ekranie graficznym; dodatkowo mogą być wyświetlane skojarzone dane alfanumeryczne (także na ekranie graficznym), np. zarys miasta i jego nazwa oraz liczba ludności,
- wszelkie dane związane z danym obiektem graficznym wyświetlanym na ekranie mogą być podane w postaci tekstowej,
- należy zapewnić łatwy dostęp do schematu z deklaracjami, definicjami i zasadami spójności,
- wszelkie dostępne na ekranie dane graficzne mogą być modyfikowane, obiekty graficzne mogą mieć zmieniane położenie lub kształt za



pomocą odpowiednich urządzeń zewnętrznych (myszka, digitizer), a związane z nimi dane alfanumeryczne mogą zostać zmienione bezpośrednio z klawiatury.

Cechy funkcjonalne języka

Język dostępu i operowania danymi w bazie graficznej musi spełniać wymagania stawiane uniwersalnym systemom graficznym. Operacje, jakie powinien realizować taki system, można podzielić na następującą grupę [6]:

1. Operowanie obrazem

- przesuwanie obrazu na ekranie,
- obrót obrazu,
- powiększenie pionowe, umożliwiające oglądanie obrazu o różnym stopniu ogólności (szczegółowe przybliżenie – ogólny obraz),
- powiększenie poziome, umożliwiające oglądanie tylko pewnych elementów obrazu wynikających z przyjętych kryteriów,
- powiększenie diagonalne, umożliwiające oglądanie części z wielu obrazów jednocześnie, wybranych na podstawie ustalonych kryteriów,
- nakładanie jednego obrazu na drugi,
- maskowanie części obrazu na podstawie ustalonych kryteriów,
- projekcja obrazu.

2. Rozpoznawanie i analiza obrazu

- rozpoznawanie zawartych w rysunku obiektów graficznych na podstawie cech opisujących ustalony wzorzec, opisany projekt nie obejmuje tego rodzaju operacji (stanowią one temat na oddzielne opracowanie).

3. Operacje przestrzenne i geometryczne

- wyznaczanie odległości między obiektami graficznymi,
- wyznaczanie długości, środka i kąta nachylenia linii,
- wyznaczanie powierzchni, środka i obwodu regionu,
- wyznaczanie przecięcia obiektów graficznych,
- wyznaczanie połączenia dwóch regionów,
- wyznaczanie różnicy między dwoma regionami.

4. Operacje funkcyjne

- funkcje agregujące (minimum, maksimum, suma, licznik),
- funkcje statystyczne (średnia, odchylenie standardowe itp.).

5. Funkcje definiowalne i połączenie z językiem proceduralnym

- możliwość definiowania własnych funkcji, dzięki zintegrowaniu języka zapytań graficznej bazy danych z językiem proceduralnym.

6. Operacje wejścia-wyjścia

- drukowanie i wyświetlanie informacji w postaci tabelarycznej,
- drukowanie i wyświetlanie obrazu graficznego,
- kolorowanie części obrazu na podstawie ustalonych kryteriów,
- zmienianie i uaktualnianie wartości ustalonych zmiennych obrazu,
- zapamiętywanie nowych informacji skojarzonych z obrazem w bazie danych.

JĘZYK SYSTEMU GT-HOLMES

System GT-HOLMES ma wszystkie cechy dedukcyjnej bazy danych i charakteryzuje się jednolitym opisem obiektów oraz zachodzących między nimi związków. Każdy obiekt jest opisany przez swój typ. Relacje, typy i obiekty są jednoznacznie identyfikowane przez swoje nazwy. Każdy obiekt może wchodzić w skład wielu relacji. Każda relacja może być widziana jako podzbiór produktu kartezjańskiego określonego na zbiorach obiektów. Elementarnymi jednostkami tego produktu są fakty, określające relacje zachodzące między konkretnymi obiektami. W tym ujęciu zbiór faktów elementarnych stanowi zawartość bazy danych systemu. Schemat bazy danych zawiera deklaracje i definicje typów, relacji oraz zasady spójności. Można powiedzieć, że schemat stanowi dla bazy danych teorię określającą jej strukturę, przez co zgodna ze schematem baza danych, staje się jednym z modeli dla tak określonej teorii [4].

Graficzny system GT-HOLMES jest rozszerzeniem systemu HOLMES w kierunku graficznych baz danych i umożliwia prowadzenie wnioskowania na danych graficznych. System oparto na trzech podstawowych pojęciach: obiekt, relacja i funkcja.

Język dostępu systemu GT-HOLMES można zanurzyć w języku proceduralnym, dzięki czemu stanie się możliwe zaimplementowanie bardziej wyrafinowanych i specjalizowanych metod dostępu, na przykład zapytywanie przez obiekty graficzne (ikony) [1].

W systemie ograniczonym do dwu wymiarów definiuje się następujące typy pierwotne: punkt, linia, region, rysunek (ang. *point*, *line*, *region*,

picture). Jest to minimalny zestaw graficznych typów pierwotnych. Wszystkie funkcje graficzne działają na dziedzinach tych typów. Dzięki takiemu podejściu projektowany język może być niezależny od implementacji struktury wewnętrznej typów. Opis wewnętrzny regionu może być linią opisującą obwód lub mapą bitową. Użytkownik bezpośredni będzie odizolowany od szczegółów realizacyjnych. Dla potrzeb systemu GT-HOLMES przyjęto pięć typów graficznych o następujących cechach:

POINT – punkt, jest to para liczb typu INTEGER, np. (123, 456);
LINE – linia łamana, jest to lista par liczb typu INTEGER, minimum dwie pary, np. (123, 456), (256, 600), (450, 900);
REGION – obszar, region, jest opisany za pomocą minimalnej liczby punktów tworzących obwód, czyli jest to lista par wartości typu INTEGER, minimum trzy pary;
COLOR – kolor obiektu graficznego, jest opisany przez trójkę liczb typu INTEGER oznaczających składniki RGB;
PICTURE – rysunek, obraz; jest opisany przez nazwę, może to być mapa bitowa, struktura wewnętrzna jest cechą charakterystyczną systemu.

Wprowadzenie typów graficznych wymaga dokonania pewnych zmian na poziomie języka, tj. dodania nowych typów standardowych (graficznych) oraz dodania funkcji graficznych. Dodatkowo wprowadzono wyrażenia arytmetyczne i funkcje agregujące. Zmieniono także składnię niektórych poleceń, aby spełnić założenia wyspecyfikowane jako cechy funkcjonalne języka.

Funkcje i predykaty graficzne

Jedną z głównych zmian w systemie HOLMES jest dodanie funkcji. Biorąc pod uwagę dyskusje na temat użytego modelu, funkcje mogą być traktowane jak wbudowane operacje, które na podstawie danych wejściowych przekazują wartość pewnego typu standardowego. Wyróżnia się cztery klasy funkcji:

● Funkcje geometryczne (obliczające)

DIST(typ graficzny, typ graficzny) → INTEGER,
LENGTH(LINE) → INTEGER,
AREA(REGION) → INTEGER,
PERIMETER(REGION) → INTEGER,
SLOPE(LINE) → INTEGER,

● Funkcje generowania obiektów graficznych

CENTER(typ graficzny) → POINT,
INTERSECTION(typ graficzny, typ graficzny) → typ graficzny,
UNION(typ graficzny, typ graficzny) → typ graficzny,
DIFFERENCE(typ graficzny, typ graficzny) → typ graficzny.

● Funkcje tworzące obiekty graficzne

SET POINT(dwójka liczb) → POINT,
SET LINE(minimum para dwójek) → LINE,
SET REGION(minimum trójka dwójek) → REGION,
SET COLOR(trójka liczb lub stała) → COLOR.

● Funkcje przekształcające obiekty graficzne

SHIFT(typ graficzny, INTEGER, INTEGER) → typ graficzny,
ROTATE(typ graficzny, INTEGER) → typ graficzny,
MAGNIFY(typ graficzny, INTEGER) → typ graficzny.

Ponadto do języka dodano przededefiniowane predykaty graficzne:

NORTH(typ graficzny1, typ graficzny2),
SOUTH(typ graficzny1, typ graficzny2),
EAST(typ graficzny1, typ graficzny2),
WEST(typ graficzny1, typ graficzny2),
COVERED BY(typ graficzny, REGION),
OVERLAP(typ graficzny, typ graficzny).

Predykaty tej grupy mogą bezpośrednio korzystać ze zbiorów indeksujących obiekty graficzne. Nie przesądzając konkretnej realizacji indeksowania informacji przestrzennej, można stwierdzić, iż będzie pożądana taka optymalizacja wykonania rozkazów języka HOLMES, aby można było prowadzić wyszukiwanie w bazie danych z wykorzystaniem z tej indeksacji.

Zdefiniowane funkcje i predykaty pozwalają operować obiektami graficznymi. Poniżej podano przykłady w języku GT-HOLMES, w których korzysta się z mechanizmów graficznych. Wszystkie przedstawione

przykłady odnoszą się do wspólnego schematu, przedstawionego w przykładzie pierwszym.

Przykład 1

```
PRIMARY TYPES
ROOM
HEATER

PRIMARY RELATIONSHIPS
ROOM mapped_by PICTURE
ROOM has_region REGION
HEATER mapped_by PICTURE
HEATER has_region REGION
HEATER has_size INTEGER

DERIVED TYPES
BIGROOM
COLDROOM

DERIVED RELATIONSHIPS

DEFINITIONS
BIGROOM_DEF:
FOR ALL x (x:BIGROOM IFF EXIST y,z ( x mapped_by y AND x
has_region z AND AREA(z) > 30 )
COLDROOM_DEF:
FOR ALL x (x:COLDROOM IFF EXIST x1, y, z, z1, v ( x mapped_by
y AND x has_region z AND x1 mapped_by y AND x1 has_region
z1 AND x1 has_size v AND COVERED_BY(z1, z) AND AREA(z)/v
< 1 )
CONSTRAINTS
CO1:
FOR ALL x:ROOM (EXIST y:HEATER; z,r,r2 ( x mapped_by z AND
x has_region r AND y mapped_by z AND y has_region r2 AND
COVERED_BY(r2, r))
```

Grupa 1

Polecenia tej grupy dotyczą rysunku na ekranie lub obiektów graficznych. W przypadku całego obrazu są to polecenia, które nie powinny znaleźć się na poziomie języka. Operacje w rodzaju „przesuń rysunek” lub „obróć” nie zmieniają fizycznej reprezentacji, nie mają więc odzwierciedlenia w bazie danych i jako takie są zaimplementowane na poziomie sprzęgu z użytkownikiem lub (i) języka proceduralnego. Operacje na obiektach graficznych zmieniają ich argumenty i są realizowane przez polecenia systemu GT-HOLMES.

Przykład 2

W przykładzie zmieniamy atrybuty grzejnika H234, tj. kolejno przesuwamy, obracamy i powiększamy obraz grzejnika:

```
CHANGE VALUES OF x:HEATER, y:REGION HOLDING (x EQ 'H234' AND x
has_region y)
(* przesun grzejnik w lewo o 10 *)
SETTING y=SHIFT(y,-10,0);
(* obróc grzejnik o 90 stopni w prawo *)
SETTING y=ROTATE(y,-90);
(* powiększ grzejnik dwukrotnie *)
SETTING y=MAGNIFY(y,2);
```

Aby zobaczyć grzejniki na pierwszym piętrze, o wielkości większej niż 20 żeberek, należy zadać pytanie:

```
FIND y, x AS temp1 HOLDING (x mapped by 'first floor' AND
x has region y AND x has size v WHICH > 20);
VIEW temp1; (* jest to odpowiednik powiększenia poziomego *)
```

Aby wyświetlić pokoje ciepłe w kolorze czerwonym oraz zimne w niebieskim, można wykonać następującą sekwencję poleceń:

```
FIND z, x, y, SET_COLOR('blue') AS cool HOLDING (x mapped by y AND
x has_region z AND x:COLDROOM);
FIND z, x, y, SET_COLOR('red') AS warmer HOLDING (x mapped by y AND
x has_region z AND NOT x:COLDROOM);
VIEW colder, warmer;
```

Wykorzystano tu definicję COLDROOM z poprzedniego przykładu.

Grupa 3 Przykład 3

W celu wyświetlenia pokoi z parteru wraz z ich powierzchniami, można zastosować pytanie:

```
FIND y, AREA(y), 'sqm' AS temp1 HOLDING (x mapped by
'ground floor' AND x has region y);
VIEW temp1;
```

Znalezienie obszaru wspólnego pokoi '15 1' i '15 2' realizuje polecenie:

```
FIND y,x,z,x1, INTERSECTION(y,z), 'intersection' AS temp2 HOL-
DING (x EQ '15 1' AND x has region y AND x1 EQ '15 2'
AND x1 has region z);
VIEW temp2;
(* zostanie wyświetlony obszar pokoi '15 1' i 15 2' oraz ich część
wspólna, wraz z opisami *)
```

W celu wyświetlenia wszystkich grzejników, które są zlokalizowane w odległości mniejszej niż 10 metrów od grzejnika 'H777', należy zadać pytanie:

```
FIND y,x AS temp3 HOLDING (x1 EQ 'H777' WHICH mapped by
z1 And x1 has region y1 AND x:HEATER WHICH mapped by
z1 AND x has region y AND DISTANCE (y, y1) < 10);
```

Grupa 4

Funkcje agregujące: MIN(INTEGER), COUNT(any type), SUM (any type), AVERAGE(INTEGER) itp. działają na zbiorze wartości, tj. na wyniku operacji FIND.

Przykład 4

Zadanie: „dla każdego piętra znajdź sumaryczny obszar i jego powierzchnię” jest realizowane przez sekwencję:

```
FIND x,SUM(z), SUM(AREA(z)) AS FULL AREA HOLDING
(y mapped by x AND x has region z);
VIEW FULL AREA;
```

Aby wyświetlić pomieszczenia parteru, które nie mają grzejników oraz pomieszczeń piwnicznych, można wykorzystać relacje z poprzedniego przykładu:

```
FIND x, y AS temp1 HOLDING (y mapped by 'ground floor' AND y
has region x AND FULL AREA('cellar', x1, z1) AND NOT
OVERLAP(x, x1) AND NOT EXIST x2, y2, z2 (y2 mapped by
'ground floor' AND y2 has region x2 AND y2 has_size z2 AND
OVERLAP(x2, x));
VIEW temp1;
```

UWAGI IMPLEMENTACYJNE

Przy korzystaniu z systemu należy brać pod uwagę następujące czynniki:

Typy graficzne w systemie

Jak już wspomniano, typy graficzne są opisane przez wartości niedostępne w sposób bezpośredni z poziomu języka. Argumenta relacji o tych typach są dostępne jedynie jako zamknięte obiekty. Dostęp do wartości tych obiektów może nastąpić wyłącznie przez specjalne funkcje. Tworzenie argumentów o tych typach następowałoby przez specjalne stałe tworzące, na przykład:

```
ADD FACTS 'car01' HAS POSITION POINT( ) WITH COLOR
COLOR( )
```

gdzie POINT() jest stałą tworzącą. Wykonanie tego polecenia spowoduje wyświetlenie komunikatu: „car01 HAS POSITION? WITH COLOR?”, a system będzie czekał na wprowadzenie współrzędnych punktu przy użyciu odpowiedniego urządzenia wejściowego, a następnie koloru (z klawiatury lub menu). Odpowiednie stałe tworzące mają następującą postać: POINT(), LINE(), REGION(), COLOR(), PICTURE().

Kolor i rysunek są typami związanymi z obiektami graficznymi, takimi jak punkt lub region. W proponowanym systemie występują one oddzielnie. Daje to całkowitą niezależność od konkretnego zastosowania, np. baza może dotyczyć jednego czarno-białego rysunku (bez użycia typów: kolor i rysunek) lub całego zbioru rysunków wielokolorowych. Tylko użytkownik – przez odpowiednie powiązania w relacjach – decyduje o tym, jakiego rysunku dotyczy dany obiekt graficzny. Dzięki temu uzyskano dużą uniwersalność w dostępie do informacji i możliwość jednolitego dostępu do informacji w ramach wielu rysunków. Można

więc prowadzić wyszukiwanie nie tylko w ramach jednego rysunku, ale całej bazy rysunków. Relacje między rysunkami lub ich hierarchia mogą zostać zdefiniowane na poziomie schematu bazy. Takie podejście jest bardziej uniwersalne niż prezentowane w publikacjach [1, 6, 10].

Sprzęż z użytkownikiem

Prezentacja danych graficznych jest znacznie bardziej skomplikowana niż przeglądanie danych w systemie klasycznym. Na przykład w odpowiedzi na pytanie: „znajdź wszystkie miasta z populacją większą niż 100 000 i wyświetl nazwy”, w oknie graficznym pojawiają się punkty (lub obszary) oznaczające miasta i obok nich ich nazwy. W oknie tekstowym pojawiają się rekordy z dwoma polami: nazwa miasta i punkt (lub region). Stan, w jakim znajduje się system, jest swego rodzaju stanem przeglądania: w oknie tekstowym można przeglądać kolejne rekordy i pola, w oknie graficznym można przeglądać obraz, dokonując na nim ewentualnie operacji przesunięcia, obrotu, powiększenia itp. Wszystko to razem stanowi pewien chwilowy stan systemu i jest pewnym obrazem informacji zawartych w bazie danych. Dane graficzne są jedynie dokładnym obrazem relacji zawartych w obszarze roboczym systemu, które wynikają bezpośrednio z wykonania operacji VIEW.

Relacje w przestrzeni roboczej

Wykonanie polecenia FIND powoduje znalezienie zbioru pewnej relacji tymczasowej. Relacja ta istnieje do czasu trwania sesji i może zostać zachowana na stałe poleceniem KEEP. Wykonanie polecenia VIEW na dowolnej relacji pierwotnej (np. na tej znalezionej) spowoduje, że znajdzie się ona w przestrzeni roboczej. Oznacza to, iż o ile istnieją w tej relacji obiekty graficzne, są one wyświetlane wraz z obiektami znakowymi występującymi w tej relacji. Załóżmy, że w wyniku wykonania pewnej pary poleceń FIND i VIEW, przestrzeni roboczej istnieją następujące fakty:

```
temp0(region,'P13')
temp0(region,'P17')
```

Na ekranie graficznym pojawiają się dwa regiony oznaczające rury wraz z ich opisami. Po wykonaniu pary poleceń FIND i VIEW, pojawiają się dodatkowe fakty:

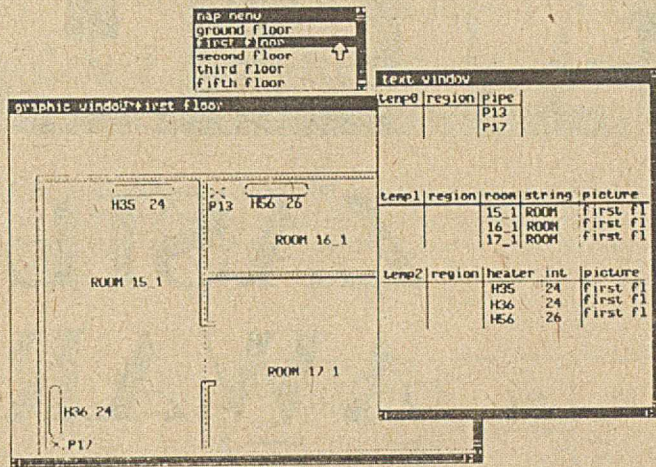
```
temp1(region,'15_1','ROOM','first_floor':picture)
temp1(region,'16_1','ROOM','first_floor':picture)
temp1(region,'17_1','ROOM','first_floor':picture)
temp2(region,'H35',24,'first_floor':picture)
temp2(region,'H36',24,'first_floor':picture)
temp2(region,'H56',26,'first_floor':picture)
```

Na ekranie ukaże się rysunek podkładowy 'first floor', poprzednie obiekty (rury) oraz regiony określające pokoje i grzejniki. Wykonanie dodatkowych poleceń FIND (i VIEW) i znalezienie wszystkich pokoi i grzejników na kolejnych piętrach spowoduje, że można oglądać wiele rysunków podkładowych z naniesionymi na nie obiektami. Użytkownik dysponuje mechanizmem (np. menu) do przeglądania wielu rysunków podkładowych. Korzystając z tych mechanizmów można zażądać wyświetlenia, na przykład, rysunku 'ground floor' parter. Wraz z tym rysunkiem pojawi się zarys pokoi i grzejników na tej kondygnacji. W dalszym ciągu widać na ekranie zarysy obu rur z faktów zawartych w relacji temp0, które nie są związane z żadnym rysunkiem podkładowym (rys. 2).

W ogólnym przypadku przewiduje się, że polecenie VIEW może być uzupełnione specyfikacją wyświetlenia obrazu składającego się z dowolnej kompozycji obrazów i relacji znajdujących się w obszarze roboczym. W razie braku takiej specyfikacji działa pewien standardowy mechanizm traktowania typu PICTURE. W szczególności, jeśli w danym fakcie występującym w przestrzeni roboczej, jeden z argumentów jest typu PICTURE, to wszystkie następne argumenty występujące w tej relacji są związane z tym właśnie rysunkiem podkładowym. Brak obiektu typu PICTURE w relacji jest równoznaczny z wyświetleniem obiektów graficznych tej relacji; bez względu na bieżący rysunek podkładowy (tj. w sposób ciągły do chwili usunięcia danego faktu z przestrzeni roboczej). Podobny mechanizm, jak dla typu PICTURE, jest związany z kolejnością wystąpień innych typów graficznych. Jeśli w fakcie wystąpi typ graficzny, to wszystkie następujące po nim typy alfanumeryczne (oraz typ kolor), aż do następnego typu graficznego, są wyświetlane wraz z danym obiektem. Na przykład fakt:

```
pipes(region1, 'P12 - 23', 'green':COLOR, region2, 'P13 - 34', 'red':COLOR)
```

spowoduje wyświetlenie regionu pierwszej rury z opisem 'P12 - 23' w kolorze zielonym i regionu drugiej rury z opisem 'P13 - 34' w kolorze czerwonym (na jednym rysunku).



Rys. 2.

Jak już stwierdzono, język systemu GT-HOLMES powstał przez dodanie typów graficznych (i związanych z nimi funkcji i rozszerzeń składni języka) do języka systemu HOLMES. Wprowadzając te rozszerzenia starano się zachować jak najwięcej z uniwersalności poprzedniego języka. Powstał w ten sposób język zgodny z poprzednim modelem i zachowujący wszystkie jego właściwości (podejście obiektowo-relacyjne i dedukcyjność), mający jednocześnie możliwości operowania danymi graficznymi. Przedstawione podejście jest swego rodzaju pośrednią drogą między dwoma innymi rozwiązaniami.

Pierwsze z nich można określić jako GO-HOLMES, co oznacza rozszerzenie języka systemu HOLMES o obiekty graficzne. Obiekty typu PRIMARY definiowane w schemacie mają, oprócz nazwy, wbudowane argumenty charakterystyczne dla opisu obiektów graficznych. Argumenty te opisują graficzną postać obiektu, a więc: obszar, kolor, rysunek podkładowy. Wartości tych argumentów są w sposób automatyczny interpretowane przez system podczas wyświetlania.

Większość poleceń systemu GO-HOLMES nie wymaga żadnych zmian w stosunku do swoich pierwowzorów z systemu HOLMES. Podzbiór obiektów znalezionych poleceniem FIND może być wyświetlany poleceniem VIEW w sposób natychmiastowy, ponieważ z każdym znalezionym obiektem graficznym są związane jego wewnętrzne argumenty reprezentujące kształt, kolor i rysunek podkładowy. Takie podejście upraszcza znacznie sposób interpretowania przez system danych graficznych. Z drugiej strony struktura obiektu jest sztywna i zaszyta na stałe w systemie.

System GO-HOLMES ma prostszy i bardziej czytelny sposób interpretowania obiektów graficznych, a co za tym idzie, całej interakcji z użytkownikiem, kosztem utraty części uniwersalności systemu opartej na typach graficznych.

Drugie rozwiązanie można nazwać GC-HOLMES rozszerzony o polecenia graficzne. Jest to propozycja, w której zakłada się dodanie do składni języka HOLMES standardowego typu tablicowego TAB[][][]. Typ ten obsługiwałby argumenty obiektu graficznego, tj. punkt, linię, region i kolor. To, czy pewien argument typu tablicowego oznacza określony argument graficzny, zależy od tego, w jakim miejscu składni polecenia zostanie umieszczony przez użytkownika.

Składnia większości poleceń standardowego systemu HOLMES nie wymaga żadnych zmian. Przede wszystkim należy dodać do składni mechanizmy operowania tablicami. Dodatkowo trzeba dodać polecenie SHOW, które umożliwia wyświetlanie danych graficznych na ekranie.

MEDCOM

ZASILANIE AWARYJNE

ZASILACZE BEZPRZERWOWE

naszej produkcji

UPS-200

UPS-300

produkcji tajwańskiej

UPS-500

produkcji japońskiej

GENERATORY PRĄDOTWÓRCZE różnych mocy

Biuro

Al. Ujazdowskie 26/39

00-478 Warszawa tel. 28 93 57, teleks 81 70 60

ZAKŁAD PRODUKCYJNO-USŁUGOWY

ul. Wołodyjowskiego 41

02-724 Warszawa tel.43 04 41

EO.3/90

Sieci MAP i TOP przegląd standardów i przykład aplikacji

Wraz z powstaniem idei łączenia komputerów w sieci, wiele firm komputerowych zaczęło proponować swoje, indywidualne rozwiązania oparte na produkowanym przez nie sprzęcie. Sieci te w wykonaniu różnych firm różniły się między sobą organizacją i sposobem przesyłania danych, a ponadto przeznaczone były dla sprzętu określonego typu. Utrudniało to znacznie przyłączanie do danej sieci sprzętu produkowanego przez inne firmy, wymianę oprogramowania i łączenie różnych sieci między sobą. Sytuacja taka spowodowała zainicjowanie w 1977 r. przez ISO (*International Organization for Standardization*) prac normalizacyjnych mających na celu ujednoczenie zasad przesyłania danych w sieciach. Wynikiem tych prac było przedstawienie w 1978 r. 7-warstwowej Architektury Systemów Otwartych zwanej modelem OSI/ISO RM (*Open System Interconnection, ISO Reference Model*).

Jednocześnie potrzeby użytkowników sieci wyprzedzały jednak tempo prac normalizacyjnych, prowadząc do powstania konstrukcji, które mają bezpośredni wpływ na kształt szczegółowych standardów. Takimi konstrukcjami wydają się być obecnie, ze względu na szczegółowe dopracowanie protokołów warstw wyższych, MAP (*Manufacturing Automation Protocol*) oraz TOP (*Technical nad Office Protocol*).

CHARAKTERYSTYKA STANDARDÓW MAP ORAZ TOP

MAP jest propozycją nowoczesnego standardu sieci lokalnej (LAN) dla zastosowań w zakładach przemysłowych. Standard ten został opracowany przez Koncern General Motors (GM) w obliczu zagrożenia amerykańskiego przemysłu motoryzacyjnego przez konkurencję japońską. GM zdał sobie sprawę, że nie może sprostać konkurencji bez szerokiego wprowadzenia automatyzacji wspomaganej standardowymi środkami komunikacyjnymi obejmującymi cały zakład przemysłowy (administrację, zarządzanie, proces projektowania, produkcję). Ponieważ żadne z istniejących rozwiązań LAN nie spełniało narzuconych przez GM wymagań, firma podjęła się opracowania własnego protokołu MAP, korzystając przy jego tworzeniu z prac prowadzonych przez wiele organizacji normalizacyjnych, takich jak: ISO, ANSI, CBEMA, IEEE.

Tym, czym dla zakładów przemysłowych stał się MAP, dla środowiska biurowego jest TOP. Podobnie jak MAP, protokół TOP powstał w koncernie Boeing, który również pilnie potrzebował sieci lokalnej umożliwiającej rozwiązanie jej specyficznych problemów.

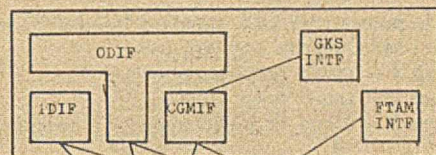
Na rys.1 przedstawiono architekturę sieci MAP i TOP wersji 3.0 na tle 7-warstwowego modelu OSI/ISO RM. W dalszej części artykułu zostaną omówione poszczególne warstwy tej wersji sieci.

Warstwa fizyczna

Warstwa fizyczna powinna zapewnić transmisję informacji przez medium transmisyjne. Dla protokołu MAP zalecanym sposobem transmisji jest transmisja szerokopasmowa o szybkości 10 Mb/s w kablu koncentrycznym stosowanym w amerykańskiej telewizji kablowej CATV. Preferowanie takiego rozwiązania wynika przede wszystkim z następujących faktów:

- Transmisja szerokopasmowa daje możliwości tworzenia kilku sieci lokalnych (o różnych funkcjach) opartych na jednym wspólnym kablu, co ma szczególne znaczenie w środowiskach przemysłowych, gdzie istnieje kilka różnych systemów łączności, umożliwiających przesyłanie informacji różnego typu (głos, dane, obraz); transmisja taka charakteryzuje się dużą niezawodnością.
- Telewizja kablowa CATV w USA opiera się na wieloletnich doświadczeniach i jest dziedziną bardzo dobrze opanowaną technicznie. Istnieje

wielu producentów zarówno kabli, jak i sprzętu pomocniczego. Eliminuje to ujemne skutki, jakie mogłyby wynikać z produkcji monopolistycznej, a dzięki produkcji masowej wpływa na relatywnie niski poziom cen.



		Jedynie MAP	MAP i TOP			Jedynie TOP
7 WARSTWA APLIKACJI	MMS ISO/DIS 9506	Network Management ISO DP 9595 ISO DP 9596	FTAM ISO 9571	Directory Services ISO DP 9594		MMS CCITT X 400
	ACSE { ISO 8649/2 /1987/ ISO 8650/2 /1987/					
6 WARSTWA PREZENTACJI	L3O 8822 /1987/ L3O 8823 /1987/ ISO 8824 ASN.1 ISO 8825 ASN.2					
5 WARSTWA SESIJ	ISO 8326 - CONNECTION ORIENTED SESSION PROTOCOL ISO 8327 - CONNECTION ORIENTED SESSION PROTOCOL					
4 WARSTWA TRANSPORTOWA	ISO 8072 /TRANSPORT SERVICE CLASS 4/ ISO 8073					
3 WARSTWA SIECI	ISO 8348 ISO 8340/AD1 ISO 8473/AD1					
	ISO/DIS 9542					X'25 PTP 35 620B
2 WARSTWA SIECI	LLC	ISO 8802.2 /1987/ TWP 1 /CLASS 1/				HDLC LAPB
	MAC	Jedynie MAP	MAP i TOP	Jedynie TOP		
		TOKEN BUS		CSMA/CD	TOKEN RING	
1 WARSTWA FIZYCZNA	IEEE 802.4 Draft1/1987 Carrierband	IEEE 802.4 Draft1/1987/ Broadband	ISO 8802.3 /1987/ Baseband Broadband	ISO 8802.5 /1987/ Baseband	TOP X'21 X 21 DIS	

Rys. 1. Architektura MAP i TOP

Należy jednak zaznaczyć, że dla MAP dopuszcza się także transmisję typu carrierband (z kodowaniem fazy) z szybkością 5 Mb/s, jak również stosowanie jako medium transmisji światłowodów. Parametry elektryczne i mechaniczne interfejsu zgodne są ze standardem IEEE 802.4.

W protokole TOP dopuszczono dużą elastyczność w doborze rozwiązań warstwy fizycznej w celu umożliwienia współpracy z sieciami różnego typu. Jednakże rozwiązaniem preferowanym jest w tym przypadku transmisja w paśmie podstawowym z użyciem kabla koncentrycznego lub skrętki zgodnie z wymaganiami standardu ISO 8802.3 lub 8802.5. Wynika to ze specyfiki środowiska biurowego opierającego się na transmisji informacji w zasadzie jednego typu danych i nie wymagającego tak wysokiej niezawodności oraz dotrzymania tak ścisłych zależności czasowych jak środowisko przemysłowe.

Warstwa połączeń logicznych

Warstwa połączeń logicznych określa format pakietów, adresację pakietów w LAN, wykrywa błędy transmisji i steruje dostępem do medium. Zgodnie z protokołem IEEE 802 dzieli się na dwie podwarstwy:

- **MAC (Medium Access Control)** – zarządzająca dostępem do łącza fizycznego (medium).
- **LLC (Logical Link Control)** – zajmująca się kontrolą przesyłania danych.

W związku z tym, że w większości procesów przemysłowych, istotne jest dotrzymanie pewnych czasów krytycznych, dla MAP wybrano realizację MAC zapewniającą zdeterminowany czas dostępu do medium – metodę przesyłania znacznika (ang. *token passing*) z wykorzystaniem topologii magistralowej (*bus*). Podstawą jest w tym przypadku standard IEEE 802.4. Odwrotna sytuacja występuje w przypadku TOP, gdzie zależności czasowe nie odgrywają takiej roli. Stąd preferuje się tu wykorzystanie standardu ISO 8802/3 (IEEE 802.3) odnoszącego się do sieci z metodą dostępu typu CSMA/CD, charakteryzująca się niezde-terminowanymi czasem dostępu.

Dla podwarstwy LLC standard ISO 8802.2 wykorzystywany przez TOP i MAC oferuje trzy typy usług:

- typ 1 – usługi bezpołączeniowe,
- typ 2 – usługi połączeniowe,
- typ 3 – usługi bezpołączeniowe z potwierdzeniem.

W przypadku usług połączeniowych (typ 2) każdy transfer jest poprzedzony ustanawianiem połączenia logicznego, w ramach którego następuje kolejowanie, potwierdzanie komunikatów, kontrola przepływu i korekta błędów. Usługi bezpołączeniowe (typ 1 i 3) nie wymagają ustanowienia połączenia logicznego.

Typ 1 usług sprowadza się właściwie tylko do wysyłania danych i nie zawiera w sobie realizacji żadnych funkcji kontrolnych związanych z transmisją. Typ 3 stanowi kategorię pośrednią między typem 1 a 2. Co prawda nie jest ustanawiane połączenie logiczne, ale każda transmisja jest potwierdzana, a także jest możliwa kontrola przepływu i retransmisja pojedynczej ramki.

Biorąc pod uwagę szybkość realizacji transmisji w MAP i TOP zaleca się wykorzystanie jedynie usług typu 1. Ponieważ nie zapewniają one jednak odpowiedniej niezawodności transmisji, większość funkcji kontrolnych realizowana jest tu przez warstwę transportową.

Warstwa sieci

Nadrzędnym zadaniem tej warstwy jest dobór trasy, którą będzie przesyłana wiadomość, tzn. adresacja węzłów znajdujących się na drodze transferu aż do węzła przeznaczenia. MAP i TOP wykorzystują zdefiniowane przez standard ISO 8348 i ISO 8340/AD1 usługi warstwy sieciowej odnoszące się do bezpołączeniowego trybu (typ 1) przesyłania informacji. Opierają się one na protokole ISO 8473 + AD1 (*Internet protocol*) opisującym sposób ich realizacji. Procedury adresacji w sieci są realizowane zgodnie ze standardami ISO 8348/AD2 i ISO/DIS 9542. ISO/DIS 8028 jest definiowany jako wewnętrznie sieciowy protokół służący do przyłączenia TOP do sieci z komutacją pakietów (X.25).

Warstwa transportowa

Warstwa transportowa powinna zapewnić przezroczysty i niezawodny transfer danych między jednostkami sesji. W związku z decyzją o realizacji usług bezpołączeniowych w warstwach niższych, MAP i TOP wykorzystują najbardziej kompleksowy i doskonały protokół zdefiniowany przez ISO dla warstwy transportowej (*Transport Protocol Class 4*). Umożliwia on, poza ustanowieniem i likwidacją połączenia, transfer danych wraz z segmentacją i łączeniem informacji, multipleksowanie połączeń transportowych, sterowanie przepływem i wielostronną kontrolę transferu zapewniającą jego niezawodność.

Warstwa sesji

Podstawowym zadaniem tej warstwy jest koordynacja komunikacji między procesami użytkowymi MAP. Usługi warstwy sesji są definiowane przez standardy ISO 8326 i 8327 w postaci jednostek funkcjonalnych (ang. *functional units*), które w fazie ustanawiania połączenia są odpowiednio grupowane w zależności od żądań użytkownika.

Dla realizacji swoich usług warstwy 5 MAP i TOP wykorzystują trzy typy nowych jednostek funkcjonalnych:

- **Kernel Functional Unit** – zapewnia podstawowe usługi wymagane do ustanowienia i likwidowania połączeń sesji i transferu danych,
- **Duplex Functional Unit** – pozwala na realizację jednoczesnej dwukierunkowej transmisji,
- **Resynchronise Functional Unit** – umożliwia uczestnikowi sesji ustalenie połączenia sesji na istniejący już lub nowy punkt synchronizacyjny.

Należy podkreślić, że *Kernel Functional Unit* jest minimalnym wymaganiem przez TOP i MAP podzbiorem standardu ISO 8326/27.

Warstwa prezentacji

Podstawową funkcją warstwy prezentacji jest konwersja i formatowanie danych. Protokoły używane na tym poziomie umożliwiają komunikującym się procesom aplikacyjnym formowanie informacji w sposób zapewniający jej rozumienie mimo różnic składniowych. MAP i TOP realizują transfer danych między warstwą sesji a poszczególnymi elementami warstwy aplikacji (*Applications Service Elements*): FTAM, MMS itd., opierając się na regułach notacji ASN. 1 (*Abstract Syntax Notation*) objętej standardami ISO 8824 i 8825.

Warstwa aplikacji

Warstwa aplikacji stanowi interfejs między użytkownikiem i siecią, przez który mają dostęp do sieci programy aplikacyjne i programiści.

Dla MAP 3.0 na poziomie tej warstwy wyróżnia się pięć protokołów:

- **Association Control Service Element (ACSE)**,
- **Directory Services** (obsługa katalogu),
- **File Transfer Access and Management (FTAM)**,
- **Network Management** (zarządzanie siecią),
- **Manufacturing Message Standard (MMS)**.

A oto krótka charakterystyka tych protokołów:

- **ACSE (ISO 8649/2, 8650/2)**. W poprzednich wersjach MAP protokół ten nazywany był CASE (*Common Application Service Elements*). Jego podstawową funkcją jest ustanawianie i likwidacja połączenia (asocjacji) między procesami użytkowymi. Pozwala na wymianę między nimi określonych parametrów, takich jak: tytuły, adresy, kontekst aplikacji. Usługi ACSE są wykorzystywane przez pozostałe protokoły warstwy aplikacji.

- **Directory Services** (obsługa katalogu). Protokół ten zajmuje się obsługą katalogu zawierającego informacje dotyczące wszystkich aktualnie dostępnych dla użytkownika zasobów sieci. Katalog zawiera nazwę każdego z zasobów wraz z adresem, przez który może być osiągalny.

- **FTAM**. Jest to protokół umożliwiający transfer zbiorów między procesami aplikacyjnymi oraz operacje na zbiorach zdalnych (tworzenie, kasowanie zbioru, transmisje zbiorów otwartych, ustawianie i zmianę atrybutów, zmianę zawartości zbioru, lokalizację, odczyt i zapis rekordów w zbiorze).

- **Network Management** (zarządzanie siecią). Protokół ten umożliwia zarządzanie siecią. Analizuje stopień wykorzystania sieci przez tworzenie statystyk dotyczących jej aktywności. Informacje te mogą być wykorzystywane np. do zmiany konfiguracji sieci i jej procesów przetwarzania, a także w celach diagnostycznych.

- **MMS**. Protokół ten ujednolica format wiadomości przesyłanych w środowisku przemysłowym. Stanowi generator wiadomości pozwalających na wzajemne porozumiewanie się różnego typu urządzeń przemysłowych (urządzeń sterowanych numerycznie, robotów przemysłowych, sterowników).

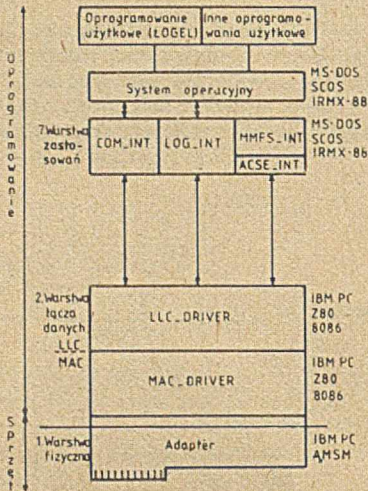
Struktura warstwy aplikacji dla TOP częściowo pokrywa się z MAP, a częściowo wykracza poza rozwiązania przyjęte przez MAP. W TOP wykorzystywane są wszystkie wymienione wyżej protokoły z wyjątkiem specjalizowanego protokołu MMS. Ponadto są stosowane:

- **Virtual Terminal (VT)**, pozwalający na implementację wirtualnego terminala,
- **Message Handling Systems (MHS)**, umożliwiający realizację poczty elektronicznej.

ram obsługi adaptera *MAC DRIVER*, który będzie zgodny ze standardem IEEE 802 na styku LLC-MAC.

Funkcje podwarstwy połączeń logicznych zgodnie z normą IEEE 802.2 (klasa 1 i 3) będzie realizował program o nazwie *LLC DRIVER*. Z chwilą, gdy zaistnieje możliwość budowy sieci miniMAP, program *LLC DRIVER* będzie mógł współpracować z adapterem sieci miniMAP, który zastąpi adapter ARCNET wraz z programem *MAC DRIVER*.

Oprogramowanie *LLC DRIVER* oraz *MAC DRIVER* powinno być wykonane w trzech wersjach dla trzech mikroprocesorów systemów: Z80/SCOS, 8086/IRMX-88 oraz IBM PC/MS-DOS.

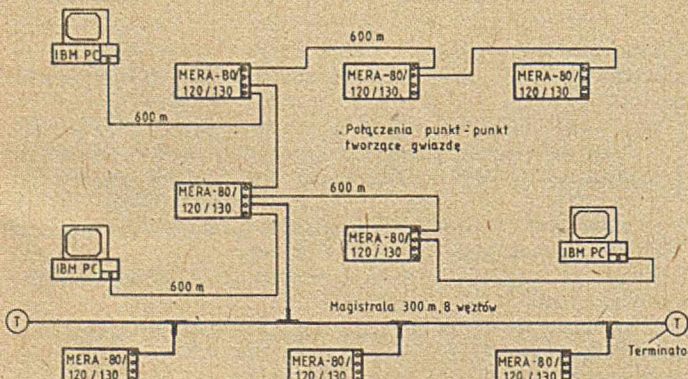


Rys. 5. Elementy sprzętowe i programowe sieci NETISS

Z usług warstwy łącza danych będą mogły korzystać następujące programy warstwy zastosowań:

LOG INT, realizujący dwukierunkowy transfer pakietów między programami użytkowymi napisanymi w języku LOGEL,
COM INT, zapewniający wymianę pakietów danych między dowolnymi, pracującymi w sieci, systemami operacyjnymi,
MMFS INT, realizujący transfer wiadomości przemysłowych na podbudowie *ACSE INT /ISO 8649/8650/*.

Przykładową konfigurację sieci NETISS przedstawiono na rys. 6.



Rys. 6. Przykładowa konfiguracja sieci NETISS

Zadaniem warstwy LLC (*Logical Link Control*) jest zapewnienie niezawodnej transmisji danych między węzłami sieci. W pełnym modelu siedmiowarstwowym ISO/OSI świadczy ona usługi warstwie NET, natomiast w przypadku sieci NETISS – warstwie aplikacji (USER).

Warstwa LLC oferuje warstwie wyższej typy usług:

- bezpołączeniowe,
- połączeniowe,
- bezpołączeniowe z potwierdzeniem.

W sieci NETISS są realizowane tylko usługi bezpołączeniowe i bezpołączeniowe z potwierdzeniem.

Usługi bezpołączeniowe polegają na przesyłaniu danych bez ustanowienia połączenia logicznego między komórkami warstwy nadrzędnej. Nie istnieje zatem mechanizm kontroli poprawności przesyłu danych i podejmowania akcji wydobywania się z błędów.

Usługi bezpołączeniowe z potwierdzeniem pozwalają przysłać dane bez ustanowienia połączenia logicznego, przy czym transmisja jest zabezpieczona przed powtórny przyjęciem tej samej ramki, a o odebraniu jednostki informacji przez zdalną komórkę LLC jest powiadamiana lokalna komórka LLC.

Znajdujące się w różnych stacjach komórki warstwy nadrzędnej w stosunku do warstwy LLC komunikują się ze sobą przez wymianę jednostek danych nazywanych LSDU (*Link Service Data Unit*).

Komórki warstwy LLC komunikują się za pośrednictwem jednostek danych PDU (*Protocol Data Unit*).

Funkcje warstwy LLC dokładnie określa norma IEEE 802.2. Najważniejsze z nich to:

- przesłanie jednostki danych (LSDU) z warstwy nadrzędnej do zdalnej stacji,
- przekazanie warstwie nadrzędnej jednostki danych odebranych ze zdalnej stacji.

Jednostki danych LLC PDU przesyłane między warstwami LLC i MAC

	Komenda	Odpowiedź
Typ 1	UI XID	XID
Typ 2	TEST ACO ACI	TEST ACO ACI

Oznaczenia:

- UI – nienumerowana jednostka danych LLC PDU,
- XID – przekazywanie informacji o parametrach warstwy LLC,
- TEST – testowanie zdolności zdalnej LLC do transmisji pola informacyjnego,
- ACO – jednostka danych LLC PDU z numerem sekwencyjnym 0,
- ACI – jednostka danych LLC PDU z numerem sekwencyjnym 1

Realizując usługi dla warstwy nadrzędnej, LLC korzysta z usług warstwy MAC. Polegają one na:

- wysłaniu do zdalnej stacji jednostki danych PDU otrzymanych z lokalnej komórki LLC,
- odbieraniu z medium transmisyjnego ramek adresowanych do stacji i przekazaniu zawartych w nich danych do lokalnej komórki LLC,
- informowaniu lokalnej komórki LLC o pomyślnym lub niepomyślnym wysłaniu ramki do medium transmisyjnego.

Warstwa nadrzędna komunikuje się z podwarstwą LLC przez wymianę tzw. prymitywów. Prymityw jest informacją określającą elementarne zdarzenie jakie może wystąpić na styku pomiędzy dwiema warstwami. Istnieją następujące prymitywy wymieniane pomiędzy warstwą nadrzędną a LLC.

• Usługi bezpołączeniowe:

- L DATA REQUEST* – zadanie przesłania danych LSDU do zdalnej stacji w trybie bezpołączeniowym.
- L DATA INDICATION* – przekazanie warstwie nadrzędnej (USER) danych LSDU otrzymanych ze zdalnej stacji w trybie bezpołączeniowym.

• Usługi bezpołączeniowe z potwierdzeniem:

- DL DATA ACK REQUEST* – zadanie wysłania LSDU do odległego LLC, przy użyciu trybu bezpołączeniowego z potwierdzeniem.
- DL DATA ACK INDICATION* – przekazanie warstwie nadrzędnej (USER) danych LSDU, nie będących duplikatem ostatnio przyjętych danych, otrzymanych ze zdalnej stacji.
- DL DATA ACK STATUS INDICATION* – informuje warstwę nad-

dokończenie na s. 16

Konwersja danych między pakietami micro CDS/ISIS i dBase III Plus

Powszechne zastosowanie mikrokomputerów w przedsiębiorstwach, placówkach naukowo-badawczych, ośrodkach informacji naukowo-technicznej i innych instytucjach spowodowało zwiększenie zapotrzebowania na różnorodne systemy zarządzania bazami danych, dostosowane do konkretnych potrzeb użytkowników. Do najpopularniejszych pakietów zarządzania bazami danych należy obecnie pakiet dBase III Plus, nadający się najlepiej do tworzenia systemów w działach kadr, w gospodarce materialowej lub księgowości. Z uwagi na brak odpowiedniego oprogramowania stosuje się go również do obsługi bibliograficznych baz danych, katalogów bibliotecznych itp., choć do tych zadań jest on gorzej dostosowany.

Wprowadzenie na rynek pakietu μ ISIS¹⁾ zaprojektowanego specjalnie do obsługi bibliograficznych baz danych, umożliwiło tworzenie systemów dokumentacyjnych wyposażonych w znacznie lepsze mechanizmy wyszu-

kiwania danych niż oferowane przez systemy zrealizowane przy użyciu dBase III. Co więcej, μ ISIS nie wymaga od użytkownika pisania procedur operowania na bazie danych; wszystkie funkcje są już oprogramowane i ujęte w odpowiednie menu [1, 2], co znacznie ułatwia projektowanie systemu. Prostota obsługi i różnorodne możliwości są zachętą do wykorzystywania pakietu μ ISIS do zakładania bibliograficznych baz danych i zarządzania nimi. Postawiło to administratorów baz danych przed problemem przeniesienia danych z pakietu dBase III do baz danych założonych w formacie μ ISIS. Odbywało się to przez eksport danych z bazy dBase do pliku ASCII, po czym plik ten był programowo przekształcany do postaci ISO 2709, a następnie do postaci CDS/ISIS. Konwersja taka wymagała więc wykonania kilku następujących po sobie operacji, a ponadto napisania dość złożonego programu konwertującego plik testowy zapisany przez pakiet dBase III do postaci ISO 2709.

W Zakładzie Systemów Informacyjnych Biblioteki Głównej i OINT Politechniki Wrocławskiej opracowano oprogramowanie, które na podstawie struktury bazy danych, zapisanej przez jeden z pakietów, tworzy odpowiadającą jej strukturę dla drugiego pakietu. Następnie dokonuje konwersji danych do postaci docelowej, uwzględniając ograniczenia obowiązujące dla tej postaci. Czynność użytkownika jest zredukowana do niezbędnego minimum; wymaga się od niego jedynie podania nazwy wyjściowej baz danych i wybrania pól konwersji [6]. Zarówno struktury baz danych, jak i fizyczna postać zbiorów dla obu pakietów są różne. Opracowanie projektu i wykonanie oprogramowania wymagało przyjęcia takich rozwiązań, które zapewniłyby uzyskanie zgodności struktur baz danych oraz zmianę postaci danych.

STRUKTURY BAZ DANYCH I POSTACI PLIKÓW W PAKIETACH μ ISIS ORAZ dBASE III

W tabeli 1 zestawiono podstawowe cechy struktur baz danych w obu pakietach [1, 4].

Tabela 1. Cechy struktur baz danych w pakietach μ ISIS i dBase III

Cechy	μ ISIS	dBase III
Maksymalna wielkość pliku głównego	16 MB	
Maksymalna liczba rekordów w bazie danych	32000	1 mld
Maksymalna długość rekordu	8000 znaków	4000 znaków
Maksymalna liczba pól w rekordzie	100	128
Maksymalna długość pola	1650 znaków	254 znaki
Występowanie zdefiniowanych pól w poszczególnych rekordach	opcjonalne	obowiązkowe
Możliwość zdefiniowania podpól w polu	istnieje	brak
Możliwość zdefiniowania pól powtarzalnych	istnieje	brak
Typ przetwarzanych rekordów	zmiennej długości	stałej długości
Dopuszczalne typy pól	znakowe numeryczne alfabetyczne wzorzec	znakowe numeryczne data logiczne memo

Postać plików w pakiecie μ ISIS

Podczas konwersji danych w kierunku μ ISIS \rightarrow dBase program wykorzystuje trzy spośród plików tworzących bazę danych w pakiecie μ ISIS:

- Plik *nazwa_bazy.FDT*, zawierający definicję struktury rekordu w bazie oraz nazwy wzorców wyświetlania, formularzy wprowadzania danych i tablic wyboru pól. Podczas konwersji z pliku tego są pobierane informacje dotyczące opisu poszczególnych pól w rekordzie, wykorzystywane do tworzenia opisu pól w pliku *nazwa_bazy.DBF* (dBase III), a mianowicie:

- nazwa bazy danych,
- nazwa formularza wyświetlania rekordu,
- nazwa tablicy wyboru pól,
- kolejno definiowane pola:
 - nazwa pola 30 bajtów (każde pole w nowym wierszu),
 - wzorzec lub ograniczniki 20 bajtów,
 - 4 liczby oddzielone spacjami:
 - identyfikator pola do 3 bajtów,
 - długość pola do 4 bajtów,
 - kod typu pola:
 - 0 znaki alfanumeryczne,
 - 1 litery alfabetu,
 - 2 cyfry,
 - 3 wzorzec,
 - znacznik powtarzalności:
 - 1 pole powtarzalne,
 - 0 w przeciwnym przypadku.

- Plik *nazwa_bazy.MST*, będący głównym plikiem bazy danych. Są z niego pobierane kolejne rekordy do konwersji, a także informacje o długości poszczególnych pól w bazie (ponieważ pola te są opcjonalne, a rekordy w bazie zarządzanej przez μ ISIS mają zmienną długość). Format zapisu tego zbioru jest następujący [3]:

ETYKIETA PLIKU stanowi ją 18 początkowych bajtów pliku. Istotna jest zawartość następujących bajtów:

- 2 3 bajt liczba rekordów w pliku,
- 4 5 bajt liczba bloków zajętych przez plik MST,
- 10 11 bajt liczba bloków w pliku po przeprowadzonej reorganizacji pliku MST,
- 14 15 bajt bloki zajęte po reorganizacji i odzyskane podczas reorganizacji.

¹⁾ Oryginalna nazwa pakietu brzmi CDS/ISIS (*Mini-micro Version*) [1]. W artykule jest stosowana skrócona nazwa μ ISIS.

ETYKIETA REKORDU – ma długość 14 bajtów i występuje na początku każdego rekordu. Istotna jest zawartość następujących bajtów:

- 0-1 bajt – numer rekordu,
- 2-3 bajt – długość rekordu,
- 8-9 bajt – adres bazowy danych,
- 10-11 bajt – liczba pól w rekordzie (każde wystąpienie pola powtarzalnego jest liczone oddzielnie).

TABLICA ADRESÓW – składa się ze zmiennej liczby elementów 6-bajtowych. Liczba ta jest równa liczbie wystąpień pól w rekordzie (każde wystąpienie pola powtarzalnego jest liczone oddzielnie). Poszczególne bajty mają następujące znaczenie:

- 0-1 bajt – identyfikator pola z pliku FDT,
- 2-3 bajt – adres pola w segmencie danych,
- 4-5 bajt – długość pola w bajtach.

SEGMENT DANYCH – stanowią go zawartości pól określonych w tablicy adresów, umieszczone jedna za drugą.

Każdy rekord musi zaczynać się od parzystego adresu, więc jeśli rekord kończy się na parzystym adresie, to zostaje dodany 1 bajt o nieokreślonej zawartości. Jest on wliczany do długości rekordu w **TABLICY ADRESÓW**.

● Plik *nawa_bazy.XRF*, zawierający indeks do pliku głównego i umożliwiający lokalizację poszczególnych rekordów w pliku *MST*. Format zapisu tego pliku jest następujący [3]:

ETYKIETA BLOKU – obejmuje pierwsze 4 bajty każdego bloku zajętego przez plik. Zawierają one:

- 0-1 bajt – numer bloku w pliku *XRF*,
- 2 bajt – wskaźnik zajętości bloku:
 - 01 – blok zapelniony,
 - 00 – w przeciwnym przypadku,
- 3 bajt – 00,

POLE DANYCH – składa się z 4-bajtowych elementów. Liczba tych elementów jest równa liczbie rekordów w pliku. Elementy te zawierają:

- 0-1 bajt – numer bloku zawierającego rekord w pliku *MST*
- 2-3 bajt – przesunięcie początku rekordu w bloku.

Postać pliku *nazwa_bazy.DBF* w pakiecie dBase

Podczas konwersji danych w kierunku *dBase*→*μISIS* program korzysta z pliku *nazwa_bazy.DBF*, tworzącego bazę danych w pakiecie *dBase III*.

Z pliku tego są pobierane zarówno informacje dotyczące opisu pól w bazie, jak i dane z kolejnych rekordów. Format zapisu tego pliku jest następujący [7]:

NAGŁÓWEK – stanowią go 32 bajty na początku pliku. Istotna jest zawartość następujących bajtów:

- 0-1 bajt – wersja bazy danych,
- 2-3 bajt – data ostatniej modyfikacji bazy danych,
- 4-7 bajt – liczba rekordów w bazie danych,
- 8-9 bajt – liczba bajtów zajmowana przez tablicę adresów,
- 10-11 bajt – liczba bajtów w rekordzie.

TABLICA ADRESÓW – ma długość 32 bajty i występuje tyle razy, ile jest pól w rekordzie. Istotne są następujące informacje:

- 0-9 bajt – nazwa pola zapisana w kodzie ASCII,
- 11 bajt – kod typu pola,
- 12-15 bajt – adres pola danych,
- 16 bajt – długość pola.

SEGMENT DANYCH – stanowią go zawartości pól rekordów; rekordy oddzielone są spacją. Pola są dopełniane spacjami do maksymalnej długości. Znacznikiem końca pliku jest 1A (szesnastkowo).

ZAŁOŻENIA PROJEKTOWE SYSTEMU

Przyjęto następujące założenia projektowe systemu konwersji danych między pakietami *dBase* i *μISIS*:

- Podczas konwersji danych, na podstawie struktury źródłowej bazy

danych, program powinien tworzyć bez udziału użytkownika analogiczną strukturę docelowej bazy danych. Obie struktury powinny być identyczne, a jeżeli to nie jest możliwe, to jak najbardziej podobne.

- Użytkownik powinien mieć możliwość wyboru pól w źródłowej bazie danych, które chce przesłać do docelowej bazy danych.

- System powinien przesyłać dane ze wszystkich rekordów bazy źródłowej, przekształcając je podczas transmisji, zgodnie z utworzoną przez niego strukturą bazy docelowej.

Różnice dotyczące struktur baz danych pakietu *μISIS* i pakietu *dBase III* powodują, że nie jest możliwa konwersja danych z jednej bazy danych do bazy o **identycznej** strukturze; w związku z powyższym przyjęto następujące rozwiązania:

- Baza danych *μISIS* dopuszcza występowanie pól powtarzalnych – pole takie może wystąpić w danym rekordzie dowolnie wiele razy. Baza danych *dBase* nie akceptuje takich pól; odpowiednikiem tego rodzaju zależności między kilkoma wartościami tego samego pola w danym rekordzie, a tym rekordem jest baza dodatkowa, połączona logicznie z główną bazą przez pole łączące. Podczas konwersji danych w kierunku *μISIS*→*dBase* dla każdego pola powtarzalnego bazy *μISIS* jest tworzona jedna baza dodatkowa *dBase*, zawierająca pole zdefiniowane w bazie *μISIS* jako powtarzalne oraz jedno pole o nazwie *NR_REKORDU*, łączące logicznie bazę główną z bazą dodatkową. Pole to zostało sztucznie wprowadzone do opisu pól w bazie głównej i zawiera kolejne numery rekordów. Wprowadzenie w sposób sztuczny pola *NR_REKORDU* zapewnia, że pole to wystąpi w każdym rekordzie. Zostawienie użytkownikowi konieczności wyboru pola łączącego nie mogłoby zapewnić spełnienia powyższego warunku, gdyż pola w bazach *μISIS* są opcjonalne. Dzięki temu zostały zachowane powiązania między kilkakrotnym wystąpieniem danego pola w rekordzie a tym rekordem. Jeśli nazwa bazy głównej w *dBase* ma postać: *nazwa_bazy.DBF*, to bazy dodatkowe mają nazwy: *A_nazwa_bazy.DBF*, *B_nazwa_bazy.DBF* itd. Można utworzyć maksymalnie 25 baz dodatkowych.

- Pola zdefiniowane przez *μISIS* jako zawierające podpola są przesyłane w taki sposób, iż z każdego podpola w bazie *μISIS* jest tworzone jedno pole w bazie *dBase*. Pole to ma długość taką, jak w bazie *μISIS*. Jeśli pole mające podpola w bazie *μISIS* miało nazwę: *nazwa_pola* i ograniczniki podpól *a* i *b*, to z pola tego zostaną utworzone dwa pola w *dBase* o nazwach: *a_nazwa_pola* i *b_nazwa_pola*.

- Ponieważ w każdej z baz danych występują pola nie mające odpowiednika typu w docelowej bazie po konwersji (np. pole typu *pattern* w bazach *μISIS* i pole typu *data* w bazach *dBase*), dokonuje się następującej konwersji typów: wszystkie pola zadeklarowane jako numeryczne pozostają po konwersji numeryczne w bazie docelowej, wszystkie pozostałe natomiast są po konwersji zadeklarowane jako pola typu znakowego.

- Pola w bazie danych *μISIS* mogą mieć maksymalną długość 1650 bajtów, w bazie *dBase* natomiast pola numeryczne mają maksymalną długość 19 bajtów, a pola znakowe – 254 bajty. Podczas transmisji danych w kierunku *μISIS*→*dBase* pola przekraczające długości dopuszczalne dla bazy *dBase* są konwertowane w następujący sposób:

- długości pól w definicji struktury bazy w pakiecie *dBase* są skracane do maksymalnych dopuszczalnych długości,
- wartości pól są odpowiednio obcinane.

- W bazach *μISIS* pola są opcjonalne (mogą nie wystąpić w danym rekordzie), w bazach *dBase* natomiast pole musi wystąpić w każdym rekordzie. Jeśli w rekordzie bazy *μISIS* określone pole nie występuje, to podczas transmisji do bazy *dBase* pole takie jest tworzone i wypełniane spacjami. Podobnie, jeśli pole ma długość mniejszą niż zadeklarowana (*μISIS* tworzy rekordy zmiennej długości), to jest ono dopełniane spacjami do długości zadeklarowanej. Przy transmisji w odwrotnym kierunku pole, które w danym rekordzie bazy *dBase* jest całkowicie wypełnione spacjami pomija się. Jeżeli wartość pola nie wypełnia pola do jego maksymalnej długości, czyli jest ono dopełnione do tej długości spacjami, to podczas transmisji pomija się spacje końcowe. W przypad-

ku pól powtarzalnych, dla których są tworzone osobne bazy w dBase, jeżeli pole takie nie wystąpiło w danym rekordzie bazy μ ISIS, to w dBase nie wystąpi ono także w bazie dodatkowej.

- Maksymalna liczba pól w rekordzie bazy μ ISIS wynosi 100. Jeżeli baza danych dBase zawiera większą liczbę pól, to pola ponad maksymalną liczbę nie są przysyłane. Dzięki możliwości dokonywania wyboru pól przez użytkownika można sterować numerami pól do przesłania.

- Maksymalna liczba rekordów bazy μ ISIS wynosi 32 tys. a bazy dBase – 1 mld. Dlatego podczas konwersji dBase \rightarrow μ ISIS przesyła się maksymalnie 32 tys. rekordów.

- Podczas transmisji w kierunku dBase \rightarrow μ ISIS dodatkowe bazy danych powiązane logicznie z bazą główną są przekształcane w ten sposób iż z pól tych baz tworzy się pola powtarzalne w bazie μ ISIS. Pole takie ma długość będącą sumą długości przekształcanych pól, a nazwę taką, jak pierwszej z tych pól. Ogranicznikami podpól są kolejne małe litery alfabetu.

Użytkownik wskazuje, które pole w dodatkowej bazie jest polem wiążącym tę bazę z bazą główną. Na podstawie zawartości tego pola pozostałe pola są dołączane do odpowiednich rekordów w bazie μ ISIS, tworząc pole powtarzalne. Wskazane pole musi być wybrane do transmisji w bazie głównej i musi mieć identyczną długość i typ, co pole wiążące.

- Pola typu *memo* w bazie dBase nie podlegają konwersji do bazy μ ISIS.

- Nazwy pól w bazie μ ISIS mogą mieć maksymalnie 30 znaków, a w bazie dBase – maksymalnie 10 znaków. Stąd nazwy pól są skracane podczas transmisji μ ISIS \rightarrow dBase do 10 znaków. Nazwy pól w dBase nie mogą zawierać spacji; zastępuje się je znakiem () pauzy w dolnej linii słów.

BADANIA CZASU KONWERSJI

Do badania wykorzystano bazę danych o nazwie CZAS, której strukturę w postaci μ ISIS przedstawiono w tabeli 2. Baza ta stanowi fragment katalogu wydawnictw ciągłych znajdujących się w zbiorach Biblioteki Głównej i OINT Politechniki Wrocławskiej [5]. W kolejnych krokach przekształcano do postaci dBase coraz większą liczbę rekordów. Uzyskiwano w ten sposób bazy danych o strukturze przedstawionej w tabeli 3. Następnie utworzone bazy danych przekształcano z powrotem do postaci μ ISIS, generując bazy danych o strukturze przedstawionej w tabeli 4.

Tabela 2. Struktura bazy danych w postaci μ ISIS przed konwersją

Lp.	Nazwa pola	Długość (bajty)	Typ	Ograniczniki – wzorzec
1	NUMER KONTROLNY	11	X	
2	KRAJ	2	X	
3	JĘZYK CZASOPISMA	3	X	
4	STANOWISKO	1	X	
5	STATUS GROMADZENIA	1	N	
6	RODZAJ ZAMÓWIENIA	3	X	
7	ISSN	9	X	
8	SYGNATURA	15	X	
9	TYTUŁ CZASOPISMA	160	X	
10	PODTYTUŁ	160	X	
11	WYDAWCA/SPRAWCA	80	X	
12	WYDAWCA/DR	80	X	
13	CZĘSTOTLIWOŚĆ	20	X	
14	CZĘSTOTLIWOŚĆ (SKRÓT)	3	X	

Pierwotna baza danych w postaci μ ISIS oraz baza otrzymana po konwersji μ ISIS \rightarrow dBase i dBase \rightarrow μ ISIS mają taką samą strukturę, z wyjątkiem dodatkowego pola o nazwie *NR_REKORDU*. Różnice dotyczą jedynie nazw pól. Pole *NR_REKORDU* jest polem tworzonym podczas konwersji danych w kierunku μ ISIS \rightarrow dBase, zgodnie z założeniami projektowymi systemu.

Tabela 3. Struktura bazy danych w postaci dBase po konwersji μ ISIS \rightarrow dBase

Lp.	Nazwa pola	Typ pola	Długość (bajty)
1	NR_REKORDU	Numeric	5
2	NUMER KONT	Character	11
3	KRAJ	Character	2
4	JĘZYK CZAS	Character	3
5	STANOWISKO	Character	1
6	STATUS GRO	Numeric	1
7	RODZAJ ZAM	Character	3
8	ISSN	Character	9
9	SYGNATURA	Character	15
10	TYTUŁ CZAS	Character	160
11	PODTYTUŁ	Character	160
12	WYDAWCA/SP	Character	80
13	WYDAWCA/DR	Character	80
14	CZĘSTOTLIW	Character	20
15	CZĘSTOTLIW	Character	3

Wyniki pomiarów czasów konwersji, zamieszczone w tabeli 5, świadczą o tym, że czas transmisji danych w kierunku dBase \rightarrow μ ISIS jest znacznie dłuższy niż w kierunku odwrotnym. Różnice te są spowodowane większą złożonością procedur tworzących pliki pakietu μ ISIS (w pliku *nazwa_bazy.MST* dla każdego rekordu są tworzone tablice adresów, a dla każdego pola – opisujące je pole danych).

Tabela 4. Struktura bazy danych w pakiecie μ ISIS po konwersji dBase \rightarrow μ ISIS

Lp.	Nazwa pola	Długość (bajty)	Typ	Ograniczniki – wzorzec
1	NR_REKORDU	5	N	
2	NUMER KONT	11	X	
3	KRAJ	2	X	
4	JĘZYK CZAS	3	X	
5	STANOWISKO	1	X	
6	STATUS GRO	1	N	
7	RODZAJ ZAM	3	X	
8	ISSN	9	X	
9	SYGNATURA	15	X	
10	TYTUŁ CZAS	160	X	
11	PODTYTUŁ	160	X	
12	WYDAWCA/SP	80	X	
13	WYDAWCA/DR	80	X	
14	CZĘSTOTLIW	20	X	
15	CZĘSTOTLIW	3	X	

Badania przeprowadzono na mikrokomputerze klasy IBM PC/AT z zegarem 10 MHz i z dyskiem twardym 20 MB o czasie dostępu ok. 65 ms.

Od połowy 1989 r. dostępna jest wersja 2.3 pakietu Micro CDS/ISIS zawierająca wiele udoskonalień i rozszerzeń w porównaniu z wersją 1.0. Fizyczna struktura zbiorów w wersji 2.3. jest niekompatybilna ze strukturą zbiorów w wersji 1.0. Przenoszenie do wersji 2.3 baz danych uprzednio utworzonych i użytkowanych w wersji 1.0 jest możliwe przez konwersję zbiorów, za pomocą narzędzi wbudowanych w obie wersje pakietu.

Tabela 5. Czasy konwersji danych między pakietami dBase III \rightarrow μ ISIS

Nazwa pliku	Liczba rekordów w pliku	Czas konwersji dBase \rightarrow μ ISIS	Czas konwersji μ ISIS \rightarrow dBase
CZAS 1	80	1'50"	1'07"
CZAS 2	160	5'43"	2'05"
CZAS 3	480	18'18"	6'37"
CZAS 4	960	35'38"	13'10"
CZAS 5	2000	1h 16'30"	26'44"
CZAS 6	4000	2h 13'30"	52'48"
CZAS 7	8000	5h 03'00"	1h 48'10"

Planuje się wzbogacenie oprogramowania o następujące możliwości:

- umożliwienie użytkownikowi przestania określonego podzbioru

zbioru wszystkich rekordów (użytkownik podaje zakres rekordów, które mają ulec konwersji)

- Dokonanie konwersji pól typu *memo* z bazy dBase III do bazy μ ISIS.
- Dokonanie konwersji bazy danych do już istniejącej nowej bazy (tzn. dokonanie jedynie konwersji samych danych, bez tworzenia struktury bazy danych).
- Umożliwienie użytkownikowi wyznaczenia w strukturze bazy danych dBase pól, które po konwersji mają być podpolami określonego pola w strukturze bazy μ ISIS.

LITERATURA

- [1] CDS ISIS (Mini Micro Version). Reference Manual. Unesco 1985 (Przekład: Prace IINTE nr 68, Warszawa 1988)
- [2] Faber R., Nowicki Z. M.: Użytkowanie pakietu mikro CDS ISIS. Warszawa: IINTE 1988
- [3] Król D.: Nowe możliwości pakietu CDS ISIS w wersji na mikrokomputery IBM PC. W: 11 Międzynarodowe Sympozjum Naukowe Studentów i Młodych Pracowników Nauki. Zielona Góra, kwiecień 1989. T.2. Elektrotechnika i elektronika. Zielona Góra: WSI 1989, s. 15-20.
- [4] Królikowski Z.: Mikrokomputerowe systemy zarządzania. Informatyka, nr 4, 1988
- [5] Nowak E.: Mikrokomputerowy system opracowania czasopism. SMOC. Praca magisterska, Zakład Systemów Informacyjnych, Politechnika Wrocławska, 1989
- [6] Szczawińska D.: System konwersji danych pomiędzy pakietami mikro CDS ISIS i dBase III. Praca magisterska, Zakład Systemów Informacyjnych, Politechnika Wrocławska, 1989
- [7] Using dBase III Plus (dokumentacja pakietu dBase III Plus).

Sieci MAP i TOP przeгляд standardów i przykład aplikacji

dokończenie ze s. 12

rzędną (USER) o sukcesie lub niepowodzeniu wcześniejszego żądania transmisji.

DL REPLY REQUEST – zadanie przesłania przez inną stację uprzednio przygotowanej tam jednostki danych LSDU lub wymiany tejże jednostki z tą stacją.

DL REPLY INDICATION – wskazuje warstwie nadrzędnej, że dane LSDU zostały włączone do odpowiedzi PDU wysłanej do stacji zdalnej.

DL REPLY STATUS INDICATION – informuje warstwę nadrzędną czy wcześniejsza wymiana danych LSDU z odległą stacją zakończona została pomyślnie lub niepomyślnie.

DL REPLY UPDATE REQUEST – określa dane LSDU, które mają być utrzymywane przez lokalną LLC. Dane te będą wysyłane po otrzymaniu zadań transmisji ze zdalnych stacji.

DL REPLY UPDATE STATUS INDICATION – jest wysyłany z lokalnej LLC do warstwy nadrzędnej w celu wskazania rezultatu wcześniejszego żądania przygotowania jednostki danych (*DL REPLY UPDATE REQUEST*). Jednostki danych LLC PDU, mające charakter komend lub odpowiedzi, są przesyłane między warstwami LLC i MAC (tabela).

Format ramki LLC PDU przedstawiono na rysunku 7.

SSAP	DSAP	CONTROL	Pole informacyjne
------	------	---------	-------------------

Rys. 7. Format ramki LLC PDU

Oznaczenia:

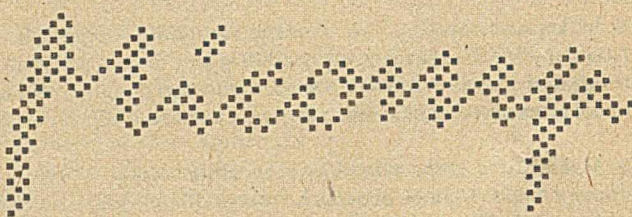
SSAP – adres nadawczej komórki warstwy nadrzędnej, unikalny tylko w obrębie stacji nadawczej.

DSAP – adres docelowej komórki warstwy nadrzędnej, unikalny tylko w obrębie stacji docelowej.

CONTROL – bajt określający typ jednostki danych oraz zawierający parametry wykorzystywane do kontroli przesyłania danych.

POLE INFORMACYJNE – pole mogące zawierać jednostkę danych warstwy nadrzędnej lub dane sterujące.

Projekt sieci NETISS stanowi nie tylko próbę zaspokojenia doraźnych potrzeb w zakresie transmisji dla zastosowań o zdeterminowanym czasie reakcji, lecz co najważniejsze otwiera drogę ku sieciom MAP i TOP z pominięciem dość istotnej bariery technologicznej, jaką jest budowa adaptera MAP Carrierband czy Broadband.



ZAKŁAD SYSTEMÓW MIKROKOMPUTEROWYCH MICOMP

40-045 Katowice, ul. Astrów 7
telefon i telefaks: 518-628
teleks: 315687

Systemy teletransmisji danych
ICL 1900, 2900, ME 29, 39,
ODRA 1305, EMCX 1305:

skaner MPXSCAN-8007,
procesor sieci WAN
MICOMP-8075
(emulacja ICL 7503),
program teletransmisji danych
MICROS FXBM
(PC, rozproszone bazy danych,
wersja sieciowa – NOVELL),
adaptery, modemy, testery,
emulatory terminali

Integracja systemów ICL i ODRA
z mikrokomputerami standardu PC
i systemami NOVELL, UNIX SCO

Instalacje w największych systemach
na terenie całego kraju!

EQ/1124/89

Edytor schematów elektronicznych

Komputerowe wspomaganie projektowania jest jednym z głównych zastosowań grafiki komputerowej. Szczególną rolę odgrywa ono w projektowaniu układów elektronicznych. Dla układów złożonych z elementów dyskretnych pozwala objąć wszystkie fazy powstawania urządzenia – rysowanie schematu, symulację działania, projektowanie ułożenia elementów na płycie i prowadzenie ścieżek. Konstruktor może w każdym z etapów pracy łatwo dokonywać poprawek – ich koszt wprowadzenia do projektu jest bardzo mały. Jeszcze większego znaczenia nabierają systemy CAD przy projektowaniu układów scalonych. Zespolecie wszystkich etapów projektowania – funkcjonalnego, logicznego, układowego, generowania masek i symulacji – pozwala uniknąć niemożliwych do poprawienia błędów projektowych. Tylko dzięki systemom CAD jest możliwy tak szybki postęp w produkcji układów scalonych.

Ogromne znaczenie oprogramowania CAD było bezpośrednią przyczyną rozpoczęcia pod koniec 1985 roku w Instytucie Informatyki Uniwersytetu Warszawskiego prac nad Edytorem schematów elektronicznych¹⁾. W trakcie realizacji projektu były wielokrotnie analizowane inne programy do rysowania schematów – PCAD, ORCAD, ACAD, REDAC-REDLOG, zwłaszcza pod kątem sposobu prowadzenia dialogu z użytkownikiem. W opracowanym edytorze wyeliminowano występujące w wymienionych programach elementy utrudniające naturalny dialog projektanta z komputerem:

- zbyt małą szybkość operacji graficznych,
 - czasochłonność wykonywania najczęściej używanych operacji – zbliżania (zoom) i przesuwania wyświetlanego na ekranie fragmentu po całym schemacie.
 - wyłącznie rysunkowe podejście do tworzenia schematu (np. ACAD i ORCAD dopuszczają nachodzenie na siebie elementów),
 - wybór elementów przez podanie nazwy (np. REDAC), a nie przez wskazanie.
- Prezentowana wersja jest wynikiem prac prowadzonych w latach 1985–1987.

Metodologia dialogu z użytkownikiem

Badania prowadzone na przełomie lat siedemdziesiątych i osiemdziesiątych, dotyczące systemów interakcyjnych, zaowocowały powstaniem pewnej ogólnej koncepcji stylu współpracy z użytkownikiem. Cechami tego jakościowo nowego podejścia do dialogu są:

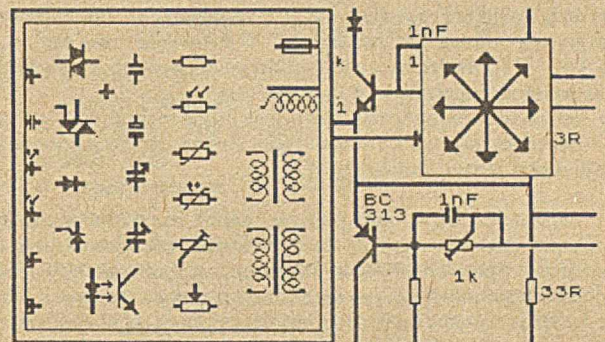
- intuicyjność i naturalność, oparte na dobrej wizualizacji (graficzne przedstawienie obiektów i wykonywanych na nich operacji) oraz na preferencji gestu (wydawanie poleceń przez wskazywanie i wybieranie symboli na ekranie),
- nieograniczanie użytkownika przez niestawianie barier w procesie interakcji (np. zmuszających go do kontynuowania rozpoczętej operacji),
- efektywność informacyjna traktowana jako możliwość symbolicznego przekazania dużej ilości informacji w prosty i czytelny dla użytkownika sposób,
- efektywność informacyjna rozumiana jako szybkość realizowania istotnych dla programu funkcji (np. wyświetlania na ekranie nowego rysunku).

Przedstawione postulaty stały się podstawą przy opracowywaniu używanego w edytorze systemu okienek, projektowaniu menu i sposobu wyświetlania na ekranie kursorów. Wykorzystanie tak zaprojektowanych narzędzi umożliwia lepszą niż w wymienionych programach współpracę z użytkownikiem.

¹⁾ Opisana wersja Edytora została opracowana w ramach prac w RPI 09. Osoby zainteresowane mogą bezpłatnie uzyskać Edytor w Instytucie Informatyki UW, PKiN, p. 848.

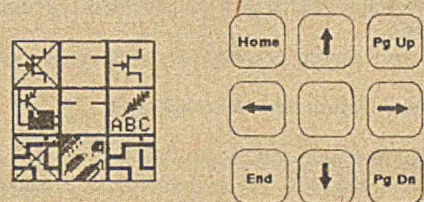
Struktury danych wyświetlanego obrazu

Edytor schematów napisano dla jedynej dostępnej w chwili rozpoczęcia prac kolorowej karty graficznej – CGA. Brak efektywnych bibliotek graficznych zmusił autora do napisania własnych procedur asemblerowych, umożliwiających szybkie wyświetlanie użytych w programie struktur danych. W wyniku przeprowadzonych prób jedynym rozwiązaniem pozwalającym uzyskać zadowalającą (w tamtych czasach) szybkość okazało się tworzenie obrazu ze zdefiniowanego w programie zestawu znaków graficznych o wymiarach 8×8 punktów. Wykorzystywanie w innych programach wyświetlania symboli z odcinków fragmentów okręgu lub liter trwałoby zbyt długo i powodowałoby rozpraszenie się użytkownika. Przykładowy wygląd ekranu w edytorze przedstawiono na rysunku 1.



Rys. 1. Przykładowy wygląd ekranu w programie

Dane są przedstawiane na ekranie w okienkach zawierających menu lub obrazy. W każdym z dziewięciu pól menu znajduje się piktogram odpowiadający pewnej funkcji lub grupie operacji edytora. Jego wybór może być dokonywany za pomocą kursora przesuwanego myszką lub przy użyciu klawiatury numerycznej (rys. 2).



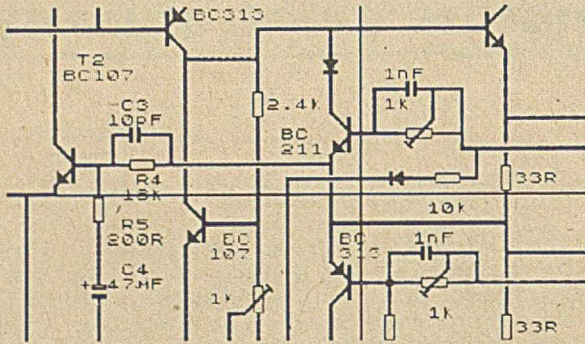
Rys. 2. Menu graficzne i odpowiadające mu klawisze

Doświadczenia wskazują na bardzo wygodne i efektywne korzystanie z takiego niestandardowego menu. Dla nowicjusza przesuwanie kursora pokazuje pole wybierane klawiszem myszki, dla doświadczonego użytkownika – położenie pola w menu łatwo kojarzy się z miejscem klawisza na klawiaturze, a więc pozwala na szybki wybór. Rysunki pól menu są pamiętane w programie w postaci map bitowych. Na czas wyświetlania menu zawartość znajdującego się pod nim fragmentu ekranu jest zapamiętywana w pamięci operacyjnej.

Obraz wyświetlany (rys. 3) w każdym z okienek jest reprezentowany w programie w postaci dwuwymiarowej tablicy rekordów, zawierających:

- kod wyświetlanego znaku,

- symetrie i obroty znaku przy wyświetlaniu (np. znak wyświetlony obrocony o 90°).
- kolor (nie dla wszystkich znaków).
- typ obiektu, który reprezentuje (np. element elektroniczny, połączenie).
- indeks w tablicy obiektów danego typu (np. w tablicy typów obiektów).



Rys. 3. Obraz na ekranie

Przedstawiony sposób organizacji ekranu umożliwia łatwy dostęp do wyświetlanych obiektów. Pozwala to na bardzo wygodną i szybką organizację dialogu z użytkownikiem. Inne udogodnienia wynikające z takiej struktury danych wyświetlanego obrazu to:

- duża szybkość wyświetlania – wyświetlane są tylko prostokątne fragmenty – łatwe do „znalezienia” w dwuwymiarowej tablicy,
- łatwość sprawdzenia, co jest w danym miejscu (np. sprawdzenia, czy coś jest pod umieszczonym tranzystorem),
- możliwość dokonywania operacji na fragmencie obrazu (np. podczas przesuwania fragmentu obrazu trzeba sprawdzić, czy nie spowoduje to zasłonięcia innych fragmentów).

Doświadczenia zebrane podczas pisania programu wykazały wiele zalet przyjętej reprezentacji danych. Dzięki takiej strukturze danych udało się wprowadzić – nie stosowaną w żadnym innym edytorze – kontrolę poprawności rysowania sieci w trakcie rysowania schematu:

- każde połączenie może zaczynać się wyłącznie od nóżki elementu; nie jest możliwe pozostawienie nie przyłączonego przewodu,
- przy rysowaniu połączeń na schemacie program sprawdza na bieżąco, czy nie zostało narysowane podwójne połączenie (cykl); wykrycie takiego połączenia powoduje jego skasowanie; można to wykorzystać do zmiany połączeń na schemacie – narysowanie nowego połączenia powoduje skasowanie starego,
- elementy i połączenia nie mogą na siebie nachodzić,
- przy przesuwaniu fragmentu schematu są zachowywane wszystkie połączenia między elementami,
- przy kasowaniu elementów ze schematu są usuwane przewody doprowadzające
- przy łączeniu dwóch przewodów. otrzymują one jednakowy kolor.

Funkcje edytora schematów elektronicznych

Operacje proponowane użytkownikowi umożliwiają rysowanie wstępnej wersji schematu oraz zmiany w trakcie realizowania projektu. Główne menu w programie dzieli wszystkie operacje na następujące grupy:

- zapisywanie i odczytywanie z dysku,
- rysowanie schematu na arkuszu głównym,
- rysowanie schematu na arkuszu pomocniczym,
- drukowanie schematu z arkusza głównego,
- zakończenie pracy programu.

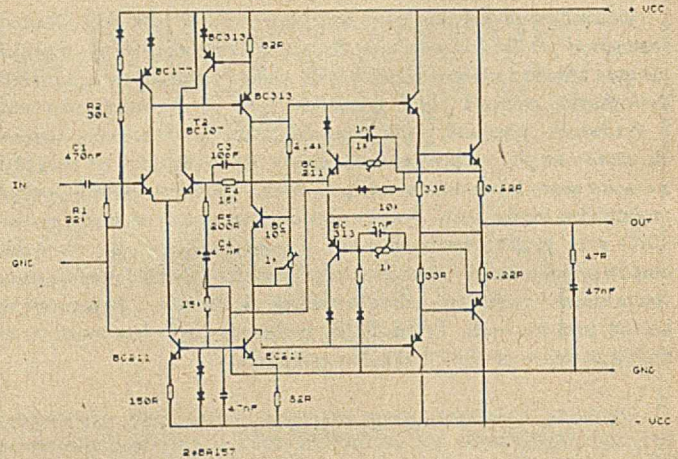
Wykorzystywane w programie operacje dyskowe umożliwiają tworzenie katalogów (np. zawierających projekt jednego urządzenia) oraz przechowywanie w nich poszczególnych schematów. Dzięki możliwości niezależnego wczytywania schematów na arkusz główny i pomocniczy schematy te można łatwo łączyć.

Operacja drukowania pozwala na wydrukowanie schematu z arkusza głównego. Jego powierzchnia odpowiada 200 × 150 znakom graficznym, czyli ok. 3300 tranzystorom (wymiary 3 × 3 znaki). Pełny rysunek

mieści się na typowym arkuszu drukarki 15-calowej. Punktowi na ekranie odpowiada punkt o wymiarach 2 × 2 na drukarce.

Na operacje rysowania schematu na arkuszu głównym i pomocniczym składają się:

- umieszczanie i kasowanie elementów,
- rysowanie i kasowanie połączeń,
- operacje wykonywane na zaznaczonym bloku: przesuwanie, powielanie, przenoszenie na drugi arkusz, kasowanie,
- zmniejszanie skali (ang. zoom), umożliwiające łatwe znalezienie interesującego fragmentu,
- umieszczanie na schemacie opisów tekstowych.



Rys. 4. Schemat wzmacniacza wykonany edytorem

Na rysunku 4 pokazano przykładowy schemat wzmacniacza małej częstotliwości wykonany przy użyciu edytora.

Doświadczenia zdobyte podczas prowadzonych prac pozwalają właściwie ocenić wykonany edytor. Na pewno wiele stosowanych w nim rozwiązań ma charakter nowatorski i dobrze byłoby zachować je w następnych wersjach. Powinien on być natomiast rozbudowany o operacje zwiększające jego walory użytkowe, tj. o stworzenie możliwości:

- definiowania przez użytkownika bibliotek elementów,
- rysowania schematu na ploterze,
- zapisywania na dysku specyfikacji sieci w formacie używanym do projektowania płytek drukowanych i symulacji.

Ze względu na znaczne zwiększenie szybkości komputerów byłaby możliwa realizacja wykorzystująca rysowanie w grafice wektorowej. Prostokątna tablica wyświetlanych znaków mogłaby zostać zastąpiona zorganizowaną w postaci grafu siatką prostokątów różnych rozmiarów. Rozwiązanie takie, przy zmniejszonym zapotrzebowaniu na pamięć, pozwoliłoby zaimplementować używaną w aktualnej wersji kontrolę poprawności wykonywanych połączeń oraz łatwiej zaprogramować przesuwanie po schemacie i przenoszenie zaznaczonych bloków.

LITERATURA

- [1] Chua L.O., Pen-Min L.: Komputerowa analiza układów elektronicznych. WNT, Warszawa 1981
- [2] Fundamental Algorithms for Computer Graphics. R. A. Earnshaw, (Ed.) Springer-Verlag, Berlin 1985
- [3] Handbook of Human Computer Interaction. M. Helander (Ed.) North-Holland, Amsterdam 1988
- [4] Hearn D., Baker M.P.: Grafika komputerowa. WNT, Warszawa 1988
- [5] Mike J. R.: The Menu Interaction Control Environment. ACM Transaction on Graphics, No. 5, 4, pp. 318-344, October 1986
- [6] Kassur A., Perkowski P.: Obliczeniowe aspekty projektowania układów elektronicznych. WNT, Warszawa 1979
- [7] Pavlidis T.: Grafika i przetwarzanie obrazów. WNT, Warszawa 1987
- [8] Perkowski P.: Technika symulacji cyfrowej. WNT, Warszawa 1980.

KOMPUTER Z WBUDOWANĄ PRZYSZŁOŚCIĄ

Komputer zaprojektowany przez czołowego światowego producenta mikrokomputerów – firmę **ADVANCED LOGIC RESEARCH** i wyprodukowany przez firmę **WEARNES TECHNOLOGY**.

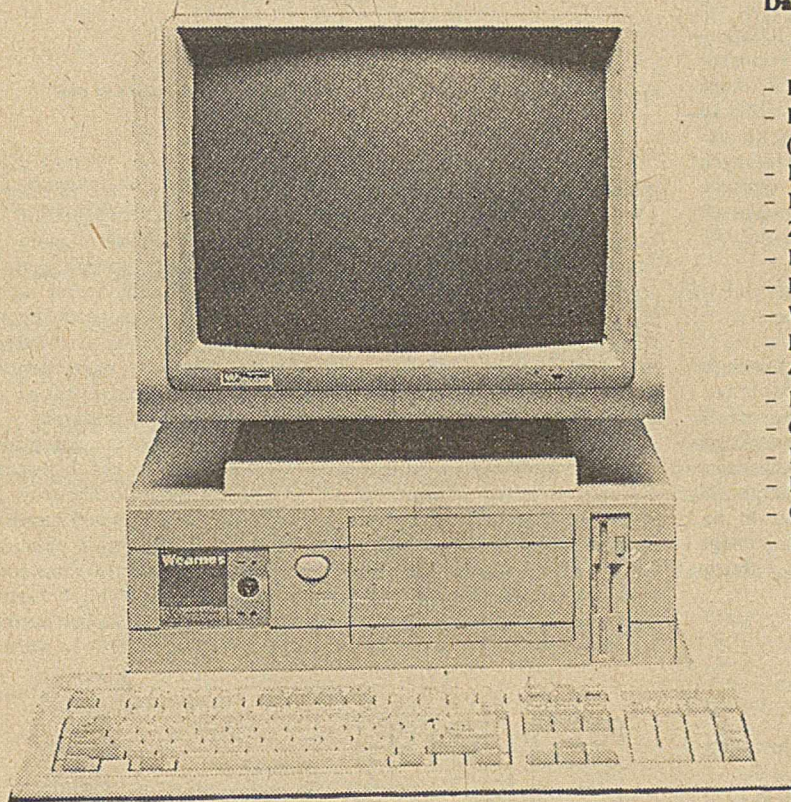
Komputer, którego możliwości i cena oszałamiają konkurencję.

Komputer, który nie zestarzeje się, dzięki możliwości rozbudowy do architektury 386SX i 486.

Dane techniczne:

SERIA WEARNES BOLDLINE „M”

- Procesor 80286 – 12.5 Mhz
- Pamięć 1 MB RAM, możliwość rozbudowy do 16 MB (5 MB na płycie głównej)
- BIOS Phoenix
- Napęd dyskietek 3,5" 1,44 MB
- Zasilacz 110 Watt
- Klawiatura 101 klawiszy
- Podstawka dla koprocatora matematycznego 80287
- Wbudowany sterownik dyskowy z przeplotem 1:1
- Port szeregowy i równoległy
- 40 MB dysk sztywny
- Możliwość korzystania z EMS 4.0
- Obudowa typu „compact”
- Miejsce na dwa napędy 5,25" o wysokości 1/2
- Podręcznik i dyskietka z programem konfiguracyjnym
- Opcjonalna rozbudowa do 386SX i 486
- 12-miesięczna gwarancja



BOLDLINE COMPUTERS
GRUPA MICOMP-TECH
Biuro Informacji Techniczno-Handlowej
ul. Astrów 7, 40-045 Katowice
telefon i telefaks: 518-628
teleks: 315687 COMP PL

DYSTRYBUTORZY:
PTH „TECHMEX”
43-300 Bielsko-Biała
ul. M. Curie Skłodowskiej 13
tel.: 42-198, 47-555, telefaks: 47-624,
teleks: 25325

Sprzętowe wzmocnianie obrazów cyfrowych w czasie rzeczywistym

Przetwarzanie cyfrowych obrazów telewizyjnych w czasie rzeczywistym (w celu np. poprawy ich jakości) wymaga zastosowania systemu cyfrowego dużej mocy obliczeniowej, np. złożonego ze znacznej liczby prostych procesorów połączonych w regularną dwuwymiarową tablicę. Z powodzeniem może być stosowany komputerowy system tablicowy (ang. *array processor*) [3, 4, 6, 7], gdyż jego struktura jest zgodna z cyfrową postacią obrazów (dwuwymiarowa tablica), a topologia połączeń między procesorami elementarnymi (lokalne połączenia między sąsiednimi procesorami tworzącymi tablicę) odzwierciedla strukturę obliczeń w operacjach lokalnych. Procesory tablicowe traktuje się często jako procesory specjalizowane. Mogą być realizowane w technologii VLSI, jak i w mniej nowoczesnych technologiach.

Podstawowymi elementami procesora tablicowego są: tablica procesorów elementarnych, układ wejścia-wyjścia, jednostka sterująca oraz sprzężenie ze współpracującym systemem komputerowym. Każdy z procesorów elementarnych jest wyposażony w pamięć lokalną i oddzielne rejestry. W wyniku postępu w technologii VLSI i UVLSI, moce obliczeniowe procesorów tablicowych o tablicach małych (16×16 pikseli) lub średnich (32×32 , 64×64 pikseli) ciągle rosną. W większości zastosowań (np. przetwarzaniu obrazów) procesory o wymienionych wielkościach tablic mają optymalne wymiary [1, 5, 6, 8]. Opisane procesory tablicowe stanowią obecnie jedną z najbardziej rozwijanych architektur nie tylko do zastosowań w przetwarzaniu obrazów, ale w ogóle w komputerach nowej generacji.

Moduł przetwarzania podobrazu

Jeżeli podobraz jest określony przez wymiar, np. 64×64 pikseli, to cały obraz (czynny) o wielkości 512×512 pikseli zawiera 64 podobrazy. W celu przetwarzania całego jednego obrazu w czasie rzeczywistym (jak w telewizji) stosuje się 64 procesory tablicowe, które działają równolegle (także w czasie rzeczywistym). Każdy procesor tablicowy stanowi jeden moduł przetwarzania podobrazu. W telewizji wykonuje się 25 obrazów na sekundę (OIRT, CCIR), tzn. każda ramka obrazowa trwa 40 ms, a każdy czynny obraz 29,9 ms. A więc czas przetwarzania jednego podobrazu (czynnego) $t_{pod} \leq 29,9$ ms, a czas przetwarzania jednego piksla

$$t_{piksl} \leq \begin{cases} 83 \text{ ns (OIRT)} \\ 100 \text{ ns (CCIR)} \\ 119 \text{ ns (FCC)} \end{cases}$$

W tabeli przedstawiono cyfrowe parametry różnych standardów obrazów telewizyjnych lub komputerowych [2].

Cyfrowe parametry różnych standardów obrazów telewizyjnych

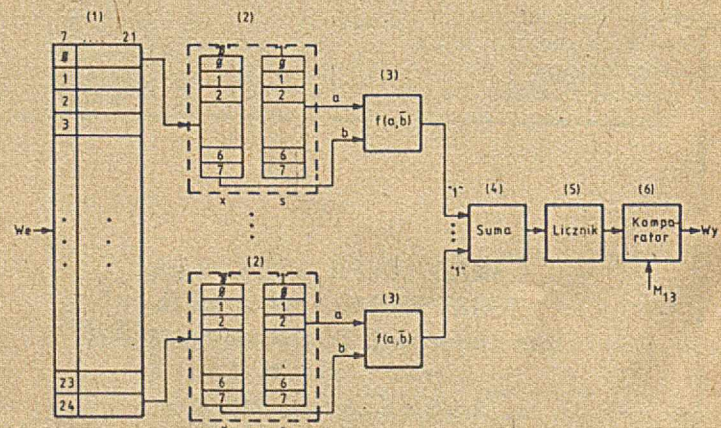
Parametr	CCIR 625 linii	FCC 525 linii	HDTV 1125 linii	Komputer
Format	4/3	4/3	5/3, 16/9	1/1
Czynne linie	575	484	1050	—
Czynne piksele/obraz	640×431	634×338	1185×998	512×512
Całkowita liczba czynnych pikseli/obraz	$0,27 \cdot 10^6$	$0,21 \cdot 10^6$	$1,18 \cdot 10^6$	$0,26 \cdot 10^6$
Wymiar piksla w poziomie/mm	0,7	0,7	0,7	—

W wypadku stosowania filtra środkowego (ang. *median filter* [9, 10]) do wzmocnienia podobrazu o wymiarze 64×64 pikseli można skonstruować sprzętowy moduł przetwarzania obrazu działającego wg schematu z rysunków 1 i 2. W celu poprawienia jakości obrazów autor stosował system zbudowany z następujących modułów: kamera telewizyjna, układ przetwornika sygnału i bufora pamięci obrazowej oraz komputer IBM PC/XT. Jako układ przetwornika sygnału zastosowano kartę typu PIP 1024A produkcji kanadyjskiej [7]. Jej zadaniem jest przekształcenie wizyjnego sygnału analogowego na wizyjny sygnał cyfrowy i utworzenie pamięci buforowej obrazu (bufora ramki obrazowej).



Rys. 1. Schemat blokowy modułu przetwarzania podobrazu o wymiarze 64×64 pikseli

Schemat na rysunkach 1 i 2 przedstawia proces przetwarzania podobrazu o wymiarze 64×64 pikseli. Sygnały cyfrowe (8-bitowe) z bufora ramki obrazowej przechodzą do detektora, który (jako układ komutacyjny) przekształca sygnał 8-bitowy na 1-bitowe sygnały dla liczników. Liczba liczników musi być równa liczbie poziomów jasności wyróżnionych na obrazie (w tym wypadku 256). Każdy licznik działa jako rejestr. Cyfrowe sygnały na wyjściu liczników są przetwarzane w układzie wykonującym obliczenia wartości histogramu (wartości poziomów jasności pikseli) i jego wartości progowej oraz wartości środkowej (filtru środkowego). Wyniki te są wprowadzone znowu do bufora ramki obrazowej: jeżeli jest stosowane okno o wielkości 5×5 pikseli, to można realizować procedurę obliczenia wartości histogramu i wartości środkowej według schematu pokazanego na rys. 2. Sygnały cyfrowe z rejestru przesuwego (1) są wprowadzane do rejestrów (2) w celu przyporządkowania wartości dwójkowej 0 lub 1 przed sumowaniem w bloku (3). Zestaw (2) i (3) zawiera 25 takich samych układów. Dane z wyjść 25 sumatorów (3) są sumowane w układzie (4) i następnie wprowadzane do licznika (5). Układ komparatora (6) wybiera wartość środkową M_{13} (dla okna o wymiarze 5×5 pikseli). Jeżeli dana z licznika jest mniejsza niż M_{13} , a suma z układu (4) jest różna od zera, to wartość

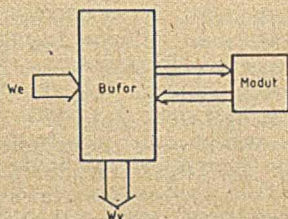


Rys. 2. Schemat blokowy realizacji procedury obliczenia wartości histogramu i wartości środkowej filtra środkowego dla okna o wymiarze 5×5 pikseli (moduł przetwarzania podobrazu); X, S – ciąg wartości 0 i 1

środkowa jest przesuwana w lewo i zwiększana o 1, do S natomiast jest ładowana poprzednia wartość S , pomnożona przez wartość X , tzn. $S \leftarrow (S \cdot X)$. Jeżeli dana z licznika jest większa niż M_{12} i suma z układu (4) jest różna od zera, to wartość środkowa jest przesuwana w lewo, a S przyjmuje wartość $(S \cdot X)$, wartość licznika natomiast jest zmniejszana o wartość sumy. Schemat taki opisuje moduł przetwarzania podobrazu i może być zastosowany w celu przetwarzania cyfrowych obrazów telewizyjnych metodą szeregową lub równoległą.

Szeregowe przetwarzanie obrazu

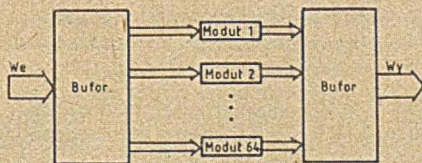
Przedstawioną procedurę można zastosować do szeregowego wzmacniania obrazu za pośrednictwem wzmacniania krawędzi obrazu. Na rysunku 3 przedstawiono schemat kolejności realizowania operacji wzmacniania obrazu. Każdy obraz jest przetwarzany szeregowo i zapamiętywany w buforze ramki. Całkowity obraz pojawia się na ekranie monitora po zakończeniu przetwarzania ostatniego podobrazu. Moduł przetwarzania podobrazu (blok (2), (3) i (4) na rys. 2) powtarza szeregowo cykl przetwarzania każdego podobrazu. Metoda ta może być stosowana w celu przetwarzania obrazu w czasie rzeczywistym, ponieważ czas przetwarzania dla całego obrazu jest większy niż czas ekspozycji jednego obrazu telewizyjnego (np. 40 ms – OIRT). W celu przetwarzania cyfrowych obrazów telewizyjnych w czasie rzeczywistym należy stosować przetwarzanie równoległe obrazu za pomocą układu wieloprocesorowego.



Rys. 3. Schemat blokowy układu przetwarzania obrazu metodą szeregową przy stosowaniu modułu przetwarzania podobrazu

Przetwarzanie obrazów w systemie wieloprocesorowym

Moduł z rysunku 2 może być stosowany dla potrzeb telewizji. Jeżeli przetwarzanie obrazu telewizyjnego (np. wzmacnianie jego krawędzi) ma się odbywać w czasie rzeczywistym, tzn. $t < 29,9$ ms, to system przetwarzania obrazu musi zawierać 64 procesory, a przetwarzanie musi odbywać się równoległe (rys. 4). W tym celu należy zastosować bardzo szybkie procesory zintegrowane w układ scalony. Realizacja takiego systemu powinna pojawić się w najbliższym czasie.



Rys. 4. 64-procesorowy system przetwarzania obrazów telewizyjnych

Metoda przetwarzania obrazów przez podział na podobrazy powoduje, że na krawędziach podobrazów pojawiają się linie czarne (brak prawidłowo wyznaczonych wartości jasności pikseli). Zniekształcenie to można usunąć przez wykonywanie obliczeń dla mniejszego okna, w ramach wcześniej wybranego większego okna. Metoda ta może być stosowana zarówno w przetwarzaniu szeregowym, jak i równoległym.

* * *

Omówione rozwiązania można podsumować w następujący sposób:

- Obrazy dwuwymiarowe wygodniej jest przetwarzać za pomocą komputera. W przyszłości może to być wykorzystane w procesie modernizacji systemów telewizji.

- Można poprawić jakość cyfrowych obrazów telewizyjnych w czasie rzeczywistym, stosując moduł przetwarzania podobrazu w systemie wieloprocesorowym oraz dostatecznie szybkie procesory.
- Sposób stosowany do poprawy jakości obrazów czarno-białych może być wykorzystywany również do poprawy jakości obrazów kolorowych, jeżeli dla każdego z kolorów podstawowych (czerwonego, zielonego i niebieskiego) będzie zawierał oddzielne przetworniki wizyjnego sygnału analogowego na wizyjny sygnał cyfrowy, np. 3 karty PIP 1024A. W tym wypadku liczba danych podlegających obróbce jest znacznie większa niż w obrazach czarno-białych. Dlatego moduł przetwarzania podobrazu musi działać odpowiednio szybko.

LITERATURA

- [1] Batcher K.: Design of a massivelley parallel processor. IEEE Trans. on Computer 1980, Vol. C-29, No. 9, pp. 836-740
- [2] Calzini M.: Hauptparameter von Kinofilm und Fernsehen sowie die Zone optimaler Sicht. Bild und Ton 1988, Nr. 3, S. 85-88
- [3] Cellary M.: Systemy wielomikroprocesorowe. Cz. I. Informatyka 1986, nr 11-12, s. 1-3
- [4] Dreyer D., Meyer K., Berger R.: Spezialprozessoren für die Echtzeit-Bildverarbeitung. Teil I, II. Bild und Ton 1989, Nr. 1, S. 19-22; Nr. 2, S. 46-50
- [5] Fisher A., Kung H.: Synchronizing large VLSI processor arrays. IEEE Trans. on Computer 1985, Vol. C-34, No. 8, pp. 734-740
- [6] Owczarczyk J., Stolarski M.: Procesory tablicowe. Współbieżne systemy przetwarzania obrazów. Informatyka 1986, nr 11-12, s. 8-11
- [7] PIP-512/1024A. Hardware manual: video digitizer board for IBM PC XT/AT. Matrox Electronic Systems Limited, Canada, March 1986
- [8] Rosenfeld A.: Parallel image processing using cellular arrays. Computer 1983, No. 1, pp. 14-20
- [9] Sach N. K.: Zmniejszanie zakłóceń obrazów za pomocą filtru medianowego. Kwartalnik Elektroniki i Telekomunikacji, PAN, 1990, nr 3
- [10] Sach N. K.: Digital 2-D picture restoration and enhancement. Kwartalnik Elektroniki i Telekomunikacji, PAN, 1990, nr 4

GT – HOLMES graficzna baza danych

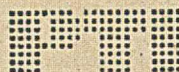
dokończenie ze s. 7

Składnia tego polecenia w pełnej postaci może mieć następującą postać: SHOW <zmienna typu pierwotnego> WITH REGION <zmienna typu tablicowego> AND COLOR <zmienna typu tablicowego> ON THE <zmienna typu ciąg znaków> HOLDING <formuła>.

Podejście to można określić jako bierne, ponieważ nie ma oczywistej zależności między informacją graficzną wyświetlaną na ekranie a informacją zawartą w bazie danych. Interpretacja danych jest całkowicie uzależniona od użytkownika.

LITERATURA

- [1] Chang N. S., Fu K. S.: Query-by-Pictorial-Example. IEEE Trans. on Software Engineering, Vol. 6, No. 6, pp. 519-524, 1980
- [2] Chang S. K., Yan C. W., Dimitroff D. C., Arndt T.: An Intelligent Image Database System. IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 681-688, 1988
- [3] Dadam P., Kuespert K., Andersen F., Blanken H.: A DBMS Prototype to Support Extended NF2 Relations: An Integrated View. ACM SIGMOD, pp. 356-367, 1986
- [4] Getta J., Rybiński H.: HOLMES – A Deduction Augmented Database Management System: Information Systems Vol. 9, No. 2, pp. 167-179, 1984
- [5] Hardwick M.: Why ROSE is Fast – Five Optimizations in the Design... ACM SIGMOD, pp. 292-238, 1987
- [6] Joseph T., Cardenas A.F.: PICQUERY – A High Level Query Language for Pictorial Database Management, IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 630-638, 1988
- [7] Kemper A., Lockemann P.C., Wallrath M.: An Object Oriented Database System for Engineering Applications. ACM SIGMOD pp. 299-310, 1987
- [8] Kirpluk M., Sobolewski P.: Implementacja dedukcyjnej bazy danych HOLMES. Informatyka, nr 2-3, 1988
- [9] Mohan L., Kashyap R.L.: An Object-Oriented Knowledge Representation for Spatial Information. IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 675-681, 1988
- [10] Roussopoulos N., Faloutsos C., Sellis T.: An Efficient Pictorial Database System for PSQL. IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 639-650, 1988
- [11] Rumbaugh J.: Relations as Semantic Constructs in an Object-Oriented Language. ACM OOPSLA, pp. 466-480, 1987
- [12] Smith K.E., Zdonik S.B.: Intermedia – A Case Study of the Differences Between Relational and Object-Oriented Database Systems. ACM OOPSLA, pp. 452-465, 1987
- [13] Stonebraker M., Rowe L.A.: The Design of POSTGRES. ACM SIGMOD, pp. 340-355, 1986.



FidoNet międzynarodowa amatorska sieć komputerowa

Historia FidoNet rozpoczęła się w 1984 roku, gdy Tom Jennings próbował wykorzystać komputer do łączności z przyjacielem zamieszkałym w drugim końcu USA – Johnem Madill. Wówczas zaczął pisać program mogący służyć jako komputerowa sekretarka – przyjmować telefony, notować listy, udostępniać niektóre informacje i transmitować pliki. Na cześć swojego wiernego przyjaciela – jamnika, Tom nazwał swój program Fido. Zaprojektował też jego portret przy użyciu podstawowego zestawu znaków ASCII. Był to prototyp tego, co obecnie nazywamy BBS (*Bulletin Board System*). Wkrótce program rozrósł się, a także krąg jego użytkowników. Fido potrafił nie tylko przyjmować telefony, ale także dzwonić – jedną godzinę w nocy wydzielono na obsługę poczty: poszczególne moduły Fido pakowały listy przeznaczone dla innych systemów i przekazywały telefonicznie do adresata. Tę formę łączności nazwano FidoNet.

Z biegiem czasu FidoNet rozrósł się obejmując swym zasięgiem pięć kontynentów (Amerykę Północną, Europę, Australię, Amerykę Południową i Afrykę) i grupując kilka tysięcy węzłów. W ten sposób FidoNet stała się największą na świecie w pełni amatorską siecią komputerową. W tym czasie powstało wiele programów obsługujących działalność sieci na każdym etapie. Wiele z nich jest na poziomie profesjonalnym, a zastosowanie niektórych (np. protokołu ZMODEM i programów go realizujących) wykracza daleko poza samą sieć.

Cechą charakterystyczną FidoNet, a w każdym razie odróżniającą ją od sieci lokalnych i wielu sieci łączących duże komputery jest to, że nie istnieje jako fizycznie trwała konstrukcja. Istnienie FidoNet to czysta umowa pomiędzy osobami biorącymi w niej udział. Nie ma trwałych łączy, wszystko odbywa się poprzez modemy i publiczną sieć telefoniczną. Dlatego też sieć nie pracuje w czasie rzeczywistym. Nie jest możliwe bezpośrednie połączenie za pośrednictwem węzła z innym węzłem. Jedyną dostępną formą łączności jest list, który można w czasie na to przeznaczonym przesłać do adresata.

FidoNet jest w tej chwili bodaj największą strukturą techniczną, która powstała bez udziału przemysłu komitetów standaryzacyjnych, zarządu, składek czy subsydiów. Pomimo prób scentralizowania zarządzania siecią, FidoNet jest w swej istocie strukturą demokratyczną, by nie rzec anarchistyczną. Znaczące zmiany w tej materii pojawiły się dopiero w zeszłym roku. Mimo formalnego nieistnienia (FidoNet nie ma osobowości prawnej) stworzono rozwiązania umożliwiające zastrzeżenie praw autorskich (Copyright) do pewnych wydawnictw. Stworzono w tym celu stowarzyszenie IFNA (*International FidoNet Association*), którego głównym zadaniem jest (a raczej było, ponieważ w chwili obecnej IFNA już nie istnieje) reprezentowanie Sieci wobec prawa.

W działalności FidoNet można wyodrębnić kilka warstw (poziomów). Warstwę pierwszą stanowi na ogół BBS i jego użytkownicy. BBS w swej najprostszej postaci, jest to program zarządzający bazą listów i bazą dostępnych dla użytkowników plików. Użytkownicy – osoby posiadające dowolny komputer z modemem i program komunikacyjny – mogą za pomocą telefonu połączyć się z BBS-em, zostawić listy lub programy dla innych użytkowników, pobrać listy adresowane do siebie lub interesujące ich pliki z dostępnej części bazy. BBS ze swej strony zapewnia pewną ochronę listów lub całych baz przed niepowołanym dostępem. Każdemu użytkownikowi przypisany jest priorytet określający czynności jakie można wykonać oraz zasoby, do których ma dostęp. Należy jednak podkreślić, że ze względu na amatorski (przynajmniej ze względu na sposób wykorzystania) charakter oprogramowania sieci, nie gwarantuje tajności i pełnego bezpieczeństwa. Nie jest to zresztą na ogół potrzebne.

Zależnie od woli i możliwości Operatora Systemu, BBS może zostać wzbogacony o wiele innych możliwości. Na przykład, często spotykaną atrakcją jest możliwość gier z innymi użytkownikami. Popularne są szachy (sprowadzające się do programu rejestrującego posunięcia graczy i analizującego ich poprawność formalną), gry w stylu „gwiazdny kupiec” itp. Znane są również gry zręcznościowe, w które można grać przez modem. Wiele BBS-ów ma rozbudowane bazy danych, dostępne dla wszystkich lub wybranych użytkowników. Istnieje możliwość zdalnego uruchamiania programów i otrzymywania przez modem wyników ich działania.

Najważniejszą jednak częścią większości BBS-ów jest baza listów. Dzięki połączeniu BBS-ów w sieć możliwości komunikacji między użytkownikami znacznie się zwiększają. W tym celu musimy jednak przejść na poziom drugi, jakim jest Sieć. Sieć (Net) stanowi grupę węzłów położonych na ogół (ze względu na koszty) w stosunkowo niewielkiej odległości. W większości przypadków są to węzły leżące w obrębie miasta lub jego dzielnicy. Węzłem sieci może być BBS, udostępniający zasoby sieci użytkownikom, lub komputer Mail Only, przeznaczony wyłącznie do wymiany poczty elektronicznej i pracy lokalnej. Węzeł wyłącznie pocztowy nie przyjmuje telefonów od użytkowników. Jeżeli strona dzwoniąca nie jest innym węzłem łączącym się w celu wymiany poczty (co jest sprawdzane za pomocą specjalnego protokołu identyfikacyjnego), połączenie jest zrywane.

Jedynym obowiązkiem węzła w stosunku do Sieci jest czuwanie o określonej porze (tak zwana Zone Mail Hour – ZMH) i odbieranie w tym czasie telefonów od innych węzłów. Ze względu na użytkowników dzwoniących do BBS-ów, a także niższe ceny połączeń, jest to pora nocna – na ogół między 2.00 a 5.00 w nocy, z uwzględnieniem różnicy pomiędzy czasem lokalnym a czasem, w którym zdefiniowano ZMH. Np. w Europie jest to godzina 3.30–4.30 czasu środkowo-europejskiego. Jest to wymóg w zasadzie historyczny, ponieważ większość węzłów pracuje jako BBS-y przez całą dobę, a stosowane obecnie powszechnie oprogramowanie odróżnia połączenia z użytkownikami od połączeń z innymi węzłami sieci. Jednak wymóg ZMH nadal istnieje, dzięki czemu można założyć węzeł FidoNet przy użyciu sprzętu wykorzystywanego do innych celów, a pracującego w charakterze węzła jedynie w nocy, oraz wykorzystanie domowego telefonu.

Sieć koordynuje obsługę poczty. Listy przeznaczone do innych węzłów tej samej sieci są na ogół wysyłane bezpośrednio do adresata, podczas gdy listy przeznaczone dla węzłów innych sieci (w ramach FidoNet) są kierowane do wyróżnionego węzła patronackiego. Pozwala to na znaczne obniżenie zarówno kosztów własnych, jak i ogólnych kosztów działania dystrybucji poczty. Węzeł patronacki rozsyła następnie listy do węzłów patronackich innych sieci lub do węzłów gromadzących pocztę do dalszego obiegu.

W Polsce istnieje jedna sieć, choć formalnie rzecz biorąc powinno być tu zastosowane nieco inne rozwiązanie ze względu na duże odległości dzielące poszczególne węzły. I najprawdopodobniej po utworzeniu większej liczby węzłów zostanie ona podzielona. Nasza sieć według używanej klasyfikacji nosi numer 480.

Kolejną warstwę stanowią **Regiony**. Region to zbiór sieci i węzłów leżących w obrębie określonego obszaru geograficznego. W Ameryce Północnej Region odpowiada na ogół kilku stanom, w Europie – poszczególnym państwom. W obecnej chwili nie istnieje regulacja obiegu na poziomie Regionów, aczkolwiek w niektórych Regionach istnieje wydzielone węzły (**Bramy**) gromadzące i rozsyłające pocztę. W zasadzie poziom Regionu został wprowadzony z dwu powodów. Jeden to konieczność zbierania danych do listy węzłów (NODELIST),

która w FidoNet jest publikowana co tydzień. Drugi – to tak zwane węzły niezależne. Otóż zdarzają się węzły leżące w znacznym (w sensie geograficznym) oddaleniu od innych węzłów, a w każdym razie poza strefą połączeń lokalnych (płatnych na ogół od liczby połączeń, a nie czasu). Próby koordynowania z nimi obiegu poczty na poziomie sieci skończyłyby się podniesieniem, zamiast obniżeniem ogólnych kosztów funkcjonowania. Takie węzły nie mogły należeć do żadnej sieci, ale dla łatwiejszej klasyfikacji są rejestrowane na poziomie Regionu. W Polsce od października 1989 istnieje osobny Region FidoNet noszący numer 48.

Kolejnym poziomem organizacji są Strefy. Strefy (ZONE) odpowiadają geograficznie kontynentom. W chwili obecnej istnieje pięć stref: 1. Ameryka Północna, 2. Europa wraz z Izraelem, 3. Australia i Oceania, 4. Ameryka Południowa oraz 5. Afryka.

Strefa ma w stosunku do reszty FidoNet dużą autonomię. Ze względu na koszty połączeń międzykontynentalnych w każdej Strefie zostały wydzielone Bramy (GATES) łączące poszczególne strefy. Większość poczty międzystrefowej jest przesyłana za pośrednictwem tychże Bram. Stosunkowo niewielka reszta jest przesyłana przez inne systemy utrzymujące z różnych przyczyn łączność międzystrefową.

Powyżej stref jest już tylko cała FidoNet. W związku z dużym rozprzestrzenieniem niezbędne są mechanizmy utrzymujące jedność i tożsamość sieci. Takimi mechanizmami, a właściwie wydawnictwami są: lista węzłów (NODELIST) publikowana w piątce oraz biuletyn FidoNews publikowany we wtorki. Oba te wydawnictwa docierają za pośrednictwem FidoNet do wszystkich węzłów sieci w ciągu kilkudziesięciu godzin. Lista węzłów jest w istocie dokumentem definiującym FidoNet, podział na Strefy, Regiony i Sieci oraz przypisującym każdemu węzłowi jednoznaczny adres, a także określającym numery telefonów związane z każdym węzłem. Ponadto w Liście zawarty jest wiele dodatkowych informacji, zarówno technicznych (podane są parametry modemu węzłów, węzły o specjalnym przeznaczeniu, jak Bramy czy *Software Distribution Systems*), jak i pozatechnicznych (np. nazwy BBS-ów związanych z węzłami, operatorów systemu, nazwiska).

Biuletyn FidoNews zawiera artykuły pisane przez użytkowników i operatorów systemu. Są tam uwagi na temat oprogramowania, artykuły, przepisy kulinarne, a nawet porady pielęgnacji zwierząt domowych. Jest to zbiorowy głos FidoNet.

Skoro już mamy opisany podmiot, to czas na zdefiniowanie przedmiotu działania, czyli czym się ta cała struktura zajmuje. Otóż podstawową działalnością FidoNet jest przekazywanie korespondencji. W obecnej chwili wyróżnić można trzy rodzaje tejże korespondencji:

- **Listy lokalne**, czyli wprowadzane w danym BBS-ie i adresowane do osób łączących się z tym BBS-em w charakterze użytkowników. Tym typem poczty nie będziemy się dalej zajmować, ponieważ FidoNet nie bierze udziału w ich przekazywaniu.
- **Net Mail**, czyli listy wprowadzane w jednym węźle i przeznaczone dla konkretnej osoby w konkretnym węźle. Z tego rodzaju poczty korzystają głównie operatorzy systemu, choć niekiedy jest ona udostępniona użytkownikom. Listy te przesyłane są metodami opisanymi poprzednio.
- **Conference Mail**, znana również pod nazwami **Echo Mail** i **Group Mail**. Te dwie ostatnie nazwy oznaczają właściwie konkretne sposoby dystrybucji tej poczty, podczas gdy pierwsza jest nazwą ogólną. Otóż w poczcie konferencyjnej zgodnie z nazwą wydzielone są pewne konferencje tematyczne. Listy umieszczane w tych konferencjach rozsyłane są do wszystkich węzłów zainteresowanych udziałem w tej konferencji. Dzięki temu w dyskusji może wziąć udział większa liczba osób. Spośród wielu istniejących konferencji część może być udostępniana użytkownikom BBS-ów, niektóre są ograniczane do węższego kręgu osób. Np. w Europie istnieje konferencja znana jako *ENET SYSOP*, przeznaczona dla operatorów systemu wszystkich europejskich węzłów FidoNet (ale nie dla użytkowników BBS-ów) oraz *INTERUSER* przeznaczona dla wszystkich zainteresowanych. Obie służą do komunikowania się na dowolne tematy. Ale jest, na przykład, konferencja *PASCAL* poświęcona wyłącznie temu językowi. Większość konferencji jest anglojęzycznych (wspólny język dla całej FidoNet), ale istnieje również konferencje dopuszczające wiele języków (*PENPAL* – również na dowolne tematy) oraz konferencje w językach lokalnych (niemieckojęzyczna *GERMAN*, czy ogólnopolska *POLECHO*).

Poczta konferencyjna to bardzo silny środek komunikacji, którego zasięg wykracza daleko poza samą FidoNet. Dość powiedzieć, że większość użytkowników stanowią ludzie nie będący operatorami systemu węzłów FidoNet, a wiedza informatyczna wielu z nich ogranicza się do umiejętności posłużenia się kilkoma programami, w tym programem komunikacyjnym.

Wymiana listów nie jest jedyną formą komunikacji w ramach FidoNet. Utworzone zostały mechanizmy umożliwiające transfer plików do wybranych węzłów (*File Attach*), czy żądanie przysłania pliku z danego węzła (*File Request*). To ostatnie może być warunkowe – transmisja następuje, jeżeli plik w posiadaniu węzła jest nowszy niż plik w posiadaniu węzła żądającego (*File Update Request*). W połączeniu z programami umożliwiającymi automatyczne rozprowadzanie plików do wybranych węzłów mechanizmy te tworzą w ramach FidoNet bardzo sprawne i szybkie narzędzie dystrybucji oprogramowania. Oczywiście szybkość jest pojęciem względnym, jednak dzięki FidoNet nowe wersje programów (które w tej klasie oprogramowania pojawiają się dość często) w przeciągu paru dni docierają do wszystkich ważniejszych odbiorców. Nowa wersja programu *QuickBBS* znalazła się w Polsce w cztery dni po opublikowaniu jej w USA. W podobny sposób rozprowadzane są np. programy antywirusowe. Jeden z popularniejszych na Zachodzie programów wykrywających *VIRUSSCAN*, identyfikujący (na dzień 5.03.1990) 67 wirusów (nie licząc mutacji) jest aktualizowany co kilka lub kilkanaście dni, w miarę pojawiania się nowych wirusów.

FidoNet nie istnieje sama dla siebie. Jej celem jest komunikacja. Byłoby więc dziwne, gdyby nie było prób połączeń FidoNet z innymi sieciami komputerowymi. Sieci takie można podzielić na dwie klasy. Pierwsza, to sieci wykorzystujące technologię FidoNet, czyli oprogramowanie oryginalnie napisane dla potrzeb FidoNet. Takich sieci (jak *ALTERNET*, *EGGNET*) jest bardzo wiele. Ze względu na wspólne formaty danych i protokoły transmisji, wymiana korespondencji na ogół nie narządza trudności. Stosuje się jeden z wariantów rozszerzonego adresowania, dzięki któremu odpowiednio inteligentne oprogramowanie może przeadresować list do odpowiedniej Bramy łączącej sieci. Trudniejsza jest sytuacja z sieciami niekompatybilnymi, jak *INTERNET*, *USENET* czy *BITNET*. Problem rozwiązano w ten sposób, że pierwszy wiersz listu musi zawierać komplet informacji umożliwiających poprawne zaadresowanie listu podczas przenoszenia do innej sieci. Po przeniesieniu list zawiera komplet informacji umożliwiającej poprawne (również automatyczne) zaadresowanie odpowiedzi.

Tak rozbudowane możliwości i ich skala nasuwają pytanie o koszty całego przedsięwzięcia. Otóż FidoNet nie jest w zasadzie dotowana jako całość. Niektóre węzły natomiast są, przy czym na ogół dotacja sprowadza się do pokrycia rachunków telefonicznych – głównego obciążenia finansowego. Większość kosztów pochodzi z węzłów odpowiedzialnych za obieg poczty. O ile więc obieg poczty i jego koncentracja obniża globalne koszty, o tyle powoduje również koncentrację miejsc powstawania tych kosztów do stosunkowo nielicznych węzłów. Ponieważ w obecnej chwili brak jest globalnych rozwiązań w sprawie podziału tych kosztów pomiędzy korzystających, więc FidoNet jest praktycznie sponsorowana przez właścicieli węzłów, przez które przepływa większość poczty.

Tak więc FidoNet jest chyba najbardziej żywiołowo rozwijającym się hobby na świecie. Bo jest to jednak hobby. Praca w sieci nie przynosi innych korzyści, niż kontakty z innymi, podobnymi ludźmi w innych krajach i nawet na innych kontynentach. Daje jednocześnie możliwość pogłębiania i rozwijania swojej wiedzy o komputerach i informatyce, a także pisanie i rozpowszechnianie własnych programów. Jest to jednak tylko działalność uboczna. Wprawdzie rozpowszechnianie oprogramowania metodą *SHAREWARE* czy *FREEWARE* zasługuje na uwagę i prawdę mówiąc powstanie FidoNet nie byłoby możliwe bez tej formy udostępnienia oprogramowania, niemniej omówienie ich wykraczałoby znacznie poza ramy niniejszego artykułu. W chwili obecnej FidoNet najbardziej przypomina mi radioamatorów lat trzydziestych. Tak samo jest to związane z dostępem do nowoczesnej techniki i pionierskich zastosowań.

Opublikowany w pierwszym numerze bieżącego roku *INFORMATYKI* artykuł p. Andrzeja Dydka „Porządkowanie zbiorów danych według polskiego alfabetu” (s. 6-7) sprowokował p. Piotra Zienkiewicza z Legnicy do nadesłania propozycji ulepszonej i uogólnionej wersji programu porządkującego. Publikując i pozostawiając do oceny zainteresowanych to rozwiązanie spodziewamy się, że pobudzi ono naszych Czytelników do nadsyłania innych pomysłów usprawniających warsztat pracy informatyka. Powszechnie wiadomo, że znakomita większość informatyków z usprawnień takich korzysta w swej codziennej pracy. Dlatego pragniemy zaapelować, aby w podobny sposób, jak to uczynił p. Zienkiewicz, również inni nasi Czytelnicy zdecydowali się za naszym pośrednictwem przekazać tego rodzaju pomysły szerszemu gronu odbiorców. Liczne przykłady podobnych zachowań spotkać można coraz częściej w wielu czasopiśmie zagranicznych.

Ulepszona wersja programu porządkującego według polskiego alfabetu

Nawiązując do artykułu „Porządkowanie zbiorów danych według polskiego alfabetu” p. Andrzeja Dydka w nrze 1/90 „Informatyki” chciałbym podzielić się ulepszoną i uogólnioną wersją programu porządkującego wg polskiego alfabetu. Moja wersja może być używana w programach aplikacyjnych budowanych przy użyciu kompilatora CLIPPER (wersja '87).

Pomysł użycia tablicy translacyjnej zastosowałem również do zamiany liter małych na duże (funkcja *PL_UPPER()*) oraz dużych na małe (funkcja *PL_LOWER()*), przy czym funkcje te prawidłowo działają również dla polskich znaków diakrytycznych.

```
# test nowych f-acji pl_order(), pl_upper(), pl_lower()
10=fopen('aaa.PL.BIN')
pl=space(32256)
fread(id,pl,32256)
fclose(id)
b=space(20)
pl_init(pl)
kl=0
do while kl<27
  clear
  @ 10,10 get b
  read
  @ 12,10 say pl_order(b)
  @ 14,10 say pl_upper(b)
  @ 16,10 say pl_lower(b)
  kl=kl+1
enddo
quit
```

Wydruk 1. Ilustracja sposobu użycia omawianych funkcji

W proponowanym przeze mnie rozwiązaniu tablica translacyjna nie jest na stałe „zaszyta” w programie, lecz może być ładowana z pliku dyskowego na początku programu aplikacyjnego. Służy do tego funkcja *PL_INIT()*. Dzięki temu przystosowanie programu aplikacyjnego do innego standardu kodowania polskich znaków narodowych (lub nawet innych języków i alfabetów) nie wymaga przeróbki i reasemblacji programu źródłowego w assemblerze, a jedynie zamiany pliku, przechowującego tablice translacyjne. W zastosowanym przeze mnie rozwiązaniu inny jest też układ kodów w tablicy – każdy ze znaków ma swój odrębny kod, dzięki czemu napisy składające się z różnych znaków dają w rezultacie wykonania funkcji porządkującej *PL_ORDER()* różne rezultaty.

Dzięki łatwości zmiany tablicy translacji można w miarę potrzeby tak ją zmieniać, aby spełniała równocześnie kilka funkcji, np. funkcję porządkującą i zamieniającą wszystkie litery małe na duże. Rozwiązanie ilustrują w sposób poglądowy zamieszczone trzy wydruki.

Wydruk 1 zawiera tekst programu źródłowego w języku CLIPPER, który ilustruje sposób użycia funkcji porządkującej *PL_ORDER()* oraz funkcji *PL_UPPER()* i *PL_LOWER()*. Tabele translacji są przechowywane w pliku *PL.BIN*, ale równie dobrze

```
CLIPPER KRYPT SYSTEM
:Source: 87
:FUNCTION: PL_UPPER, PL_LOWER, PL_ORDER, PL_INIT
:SYNTAX: PL_UPPER( <list> ), PL_LOWER( <list> ), PL_ORDER( <list> ),
          PL_INIT( <list> )
:NOTE: Inna, uogólniona wersja funkcji do indeksacji wg
       polskiego alfabetu
       Bez kontroli typu i ilości przekazywanych parametrów
       PL_INIT( <tablica> ) ładuje tablicę konwersji
       <tablica> powinna mieć długość 256 * 3 znaków
       <list> to lista dla PL_ORDER(), potem dla PL_UPPER(),
       na końcu dla PL_LOWER()

PUBLIC PL_ORDER, PL_INIT, PL_UPPER, PL_LOWER
KRYPT PARC : PAR
KRYPT PARCLEN : PAR
KRYPT SRTC : PAR
KRYPT SRTCL : PAR
KRYPT SRTCLLEN : PAR

:PROG SEGMENT CODE
ASSUME CS: PROG

TAB_TRANS DB 256 dup(?) ; tablica translacji f-acji PL_ORDER()
TAB_TRANS1 DB 256 dup(?) ; tablica dla PL_UPPER()
TAB_TRANS2 DB 256 dup(?) ; tablica dla PL_LOWER()
TAB_TLEN EQU 256*3 ; długość tabel
BUFOR DB 256 dup(0) ; bufor na wynik
POB DW 0 ; aliczba pomocnicza
TAB_OFF DW 0 ; która tablica

PL_INIT PROC FAR
push bp
mov bp,sp
push ds
push es
push si
push di
mov ax,1
push ax
call _parc
add sp,2
mov ds,ds
mov si,ax
mov ax,cs
mov es,ax
lea di,TAB_TRANS
mov cx,TAB_TLEN
cld
rep movsb ; załadowanie tabel translacji
pop di
pop si
pop es
pop ds
pop bp
mov ax,0
push ax
call _retcl
add sp,2
ret
PL_INIT ENDP

PL_ORDER PROC FAR
mov cx,TAB_OFF,0
Jmp PL
PL_ORDER ENDP

PL_UPPER PROC FAR
mov cx,TAB_OFF,256
Jmp PL
PL_UPPER ENDP

PL_LOWER PROC FAR
mov cx,TAB_OFF,512
Jmp PL
PL_LOWER ENDP

PL PROC FAR
push bp
mov bp,sp
push ds
push es
push si
push di
mov ax,1
call _parc
add sp,2
mov ds,ds
mov si,ax
mov ax,cs
mov es,ax
lea di,BUFOR
lea bx,TAB_TRANS
add bx,cs:TAB_OFF
mov cx,cs:POB
cld
lodsb
xlat byte ptr cs:(bx)
stosb
loop _retcl
mov bx,cs:POB
mov dx,cs
lea ax,BUFOR
pop di
pop si
pop es
pop ds
push bx
push ds
push ax
call _retclen
add sp,4
ret
PL ENDP

PROG ENDS
END
```

Wydruk 2. Tekst źródłowy w assemblerze

INFOSYSTEM '90

mogą być umieszczone w zmiennej znakowej przechowywanej w pliku typu *.MEM.

Wydruk 2 zawiera omawiane funkcje zapisane w postaci źródłowej w assemblerze. Zwróciłbym uwagę na pewną różnicę w stosunku do wersji p. Dydka. Mianowicie założył on, że znakowy argument funkcji nie może wewnątrz zawierać bajtów zerowych – bajt zerowy traktowany jest jako znacznik końca tekstu. Tymczasem wersja CLIPPER'87 dopuszcza stosowanie tekstów złożonych z dowolnych znaków o pełnym zakresie kodów 0..255. Wersja proponowana przeze mnie uwzględnia ten fakt.

ZAWARTOŚĆ PLIKU "PL.BIN" W POSTACI SZESNASTKOWEJ

0000	00 01 02 03	04 05 06 07	08 09 0A 0B	0C 0D 0E 0F
0010	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
0020	20 21 22 23	24 25 26 27	28 29 2A 2B	2C 2D 2E 2F
0030	30 31 32 33	34 35 36 37	38 39 3A 3B	3C 3D 3E 3F
0040	40 41 42 43	44 45 46 47	48 49 4A 4B	4C 4D 4E 4F
0050	50 51 52 53	54 55 56 57	58 59 5A 5B	5C 5D 5E 5F
0060	60 61 62 63	64 65 66 67	68 69 6A 6B	6C 6D 6E 6F
0070	70 71 72 73	74 75 76 77	78 79 7A 7B	7C 7D 7E 7F
0080	80 81 82 83	84 85 86 87	88 89 8A 8B	8C 8D 8E 8F
0090	90 91 92 93	94 95 96 97	98 99 9A 9B	9C 9D 9E 9F
00A0	AA AB AC AD	AE AF B0 B1	B2 B3 B4 B5	B6 B7 B8 B9
00B0	BA BB BC BD	BE BF C0 C1 C2	C3 C4 C5 C6	C7 C8 C9 CA
00C0	CB CC CD CE	CF D0 D1 D2 D3	D4 D5 D6 D7	D8 D9 DA DB
00D0	DC DD DE DF	E0 E1 E2 E3	E4 E5 E6 E7	E8 E9 EA EB
00E0	EC ED EE EF	F0 F1 F2 F3	F4 F5 F6 F7	F8 F9 FA FB
00F0	FC FD FE FF			
0100	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
0110	20 21 22 23	24 25 26 27	28 29 2A 2B	2C 2D 2E 2F
0120	30 31 32 33	34 35 36 37	38 39 3A 3B	3C 3D 3E 3F
0130	40 41 42 43	44 45 46 47	48 49 4A 4B	4C 4D 4E 4F
0140	50 51 52 53	54 55 56 57	58 59 5A 5B	5C 5D 5E 5F
0150	60 61 62 63	64 65 66 67	68 69 6A 6B	6C 6D 6E 6F
0160	70 71 72 73	74 75 76 77	78 79 7A 7B	7C 7D 7E 7F
0170	80 81 82 83	84 85 86 87	88 89 8A 8B	8C 8D 8E 8F
0180	90 91 92 93	94 95 96 97	98 99 9A 9B	9C 9D 9E 9F
0190	AA AB AC AD	AE AF B0 B1	B2 B3 B4 B5	B6 B7 B8 B9
01A0	BA BB BC BD	BE BF C0 C1 C2	C3 C4 C5 C6	C7 C8 C9 CA
01B0	CB CC CD CE	CF D0 D1 D2 D3	D4 D5 D6 D7	D8 D9 DA DB
01C0	DC DD DE DF	E0 E1 E2 E3	E4 E5 E6 E7	E8 E9 EA EB
01D0	EC ED EE EF	F0 F1 F2 F3	F4 F5 F6 F7	F8 F9 FA FB
01E0	FC FD FE FF			
01F0	10 11 12 13	14 15 16 17	18 19 1A 1B	1C 1D 1E 1F
0200	20 21 22 23	24 25 26 27	28 29 2A 2B	2C 2D 2E 2F
0210	30 31 32 33	34 35 36 37	38 39 3A 3B	3C 3D 3E 3F
0220	40 41 42 43	44 45 46 47	48 49 4A 4B	4C 4D 4E 4F
0230	50 51 52 53	54 55 56 57	58 59 5A 5B	5C 5D 5E 5F
0240	60 61 62 63	64 65 66 67	68 69 6A 6B	6C 6D 6E 6F
0250	70 71 72 73	74 75 76 77	78 79 7A 7B	7C 7D 7E 7F
0260	80 81 82 83	84 85 86 87	88 89 8A 8B	8C 8D 8E 8F
0270	90 91 92 93	94 95 96 97	98 99 9A 9B	9C 9D 9E 9F
0280	AA AB AC AD	AE AF B0 B1	B2 B3 B4 B5	B6 B7 B8 B9
0290	BA BB BC BD	BE BF C0 C1 C2	C3 C4 C5 C6	C7 C8 C9 CA
02A0	CB CC CD CE	CF D0 D1 D2 D3	D4 D5 D6 D7	D8 D9 DA DB
02B0	DC DD DE DF	E0 E1 E2 E3	E4 E5 E6 E7	E8 E9 EA EB
02C0	EC ED EE EF	F0 F1 F2 F3	F4 F5 F6 F7	F8 F9 FA FB
02D0	FC FD FE FF			

Wydruk 3. Zawartość pliku PL.BIN w postaci szesnastkowej

Wydruk 3 przedstawia zawartość pliku „PL.BIN”, przechowywanego w sobie trzy tabele translacji – każda licząca po 256 bajtów. Pierwsza tabela – jest brana pod uwagę przez funkcję `PL_ORDER()`, druga i trzecia – przez odpowiednio funkcję `PL_UPPER()` i `PL_LOWER()`. Tabele zostały skonstruowane dla komputera pracującego ze standardem polskich znaków wg Domu Handlowego Nauki.

Należy zwrócić uwagę, że wszystkie z powyższych trzech funkcji są wykonywane identycznie; interpretacja ich działania zależy jedynie od zawartości tabel translacji.

PIOTR ZIENKIEWICZ

W dniach 8–12 kwietnia 1990 r. odbyły się w Poznaniu kolejne Międzynarodowe Targi Elektroniki, Telekomunikacji i Techniki komputerowej INFOSYSTEM'90. Wzięło w nim udział 225 wystawców z 17 krajów, czyli nieco więcej niż w roku ubiegłym. Zwiększyła się liczba firm zagranicznych uczestniczących w Targach (z 41 w roku ubiegłym do 48), mniej było natomiast polskich firm (zwłaszcza tych pośredniczących) i prezentowały się znacznie skromniej. Z rozmachem urządzone stoiska takich wystawców, jak Soft-Tronic, Polsin i Epson przyciągały tłumy ciekawych, choć niewiele było nowości do obejrzenia.

Oferta targowa obejmowała jak zwykle przede wszystkim sprzęt komputerowy, całą gamę urządzeń peryferyjnych, podzespoły elektroniczne, urządzenia telekomunikacyjne i aparaturę kontrolno-pomiarową. Ponadto oczywiście oprogramowanie systemowe i użytkowe, coraz bardziej dostosowane do potrzeb polskiego użytkownika.

Niedawno powstała firma Kirk Electronics, będąca dystrybutorem drukarek firmy OKI oraz kopiarek i telefaksów firmy Developpe, zaprezentowała bogatą ofertę tych urządzeń. Wśród drukarek najbardziej reklamowano małą, lekką i taną drukarkę laserową OKI Laser 400, za jedyne 2499 DM. Podobnie wśród kopiarek, oprócz droższych urządzeń mających bardziej wyrafinowane możliwości, znalazła się oferta taniej, małej i prostej koparki Decelope 10, za 699 DM. Występując z tak tanimi urządzeniami (co nie oznacza gorszej jakości) firma ma nadzieję zdobyć możliwie szybko polski rynek. Jako ciekawostkę warto dodać, iż OKI jako jedyna na świecie firma oferuje drukarkę zautomatyzowaną linię produkcyjną drukarek.

Interesy firmy Kirk Electronics w Polsce prowadzi spółka Atut, która przedstawiła także własną ofertę, obejmującą m.in. maszyny bankowe do kodowania i adresowania czeków i innych dokumentów bankowych, urządzenia do liczenia banknotów oraz do okładania dokumentów. Przedstawiciel firmy Atut ubolewał nad tym, iż wielkość polskich czeków jest niestety nieodpowiednia do oferowanych przez niego urządzeń – wystarczyłoby zwiększyć szerokość czeków o kilka milimetrów, aby na szeroką skalę wprowadzić te urządzenia w naszych bankach.

Urządzenia bankowe znalazły się również w ofercie firmy Philips; reklamowano dwa rodzaje automatów bankowych do wypłat, które mogą być bezpośrednio połączone z dużymi systemami komputerowymi obsługi banków. Niestety, nie można ich było obejrzyć na targach, podobnie zresztą jak gigantycznego systemu archiwizacji MEGA DOC, zrealizowanego z użyciem dysków optycznych i przeznaczonego do obsługi baz danych w bibliotekach, bankach, muzeach, agencjach itp. (kilkadziesiąt gigabajtów w urządzeniu przypominają

jącym szafę grającą). System ten, dotychczas objęty embargiem, miał bardzo niewielkie szanse znalezienia się w Polsce, ale przedstawiciele Philipsa zapewnili, że już wkrótce sytuacja ulegnie zmianie.

To, co można było obejrzyć w stoisku Philipsa, to telefaksy i sprzęt komputerowy, zwłaszcza znakomite wielodostępne systemy minikomputerowe serii P90 x 0, z procesorem Motorola 68030 i zegarem od 25 do 50 MHz. Umożliwiają one dołączenie do 160 stanowisk, połączenia zarówno z dużymi systemami, jak i ze sprzętem klasy PC. Zapewniają pełną ochronę danych w razie zaniku zasilania, wysoką niezawodność pracy i dużą moc przetwarzania.

Już od kilku lat Rank Xerox zapowiadał wprowadzenie na rynek spolszczonej wersji popularnego już u nas systemu DTP – Ventura Publisher. W tym roku również można było usłyszeć zapewnienie, że już wkrótce polska wersja Ventury będzie na rynku – tym razem zapewniała o tym firma Pro International Ltd. Oferuje ona trzy typowe zestawy DTP w wersjach MINI, MEDIUM i LARGE (odpowiednio do potrzeb użytkowników). Zestaw najbardziej rozbudowany jest wyposażony w naswietlarkę postscriptową firmy Setter. Może ona pracować z rozdzielczością od 400 do 3000 punktów na cal, co pozwala na wybranie optymalnej rozdzielczości dla tekstu i grafiki oraz uzyskanie 256 odcieni szarości. Obecnie taki system jest używany do składania *Gazety International* (angielskojęzycznej wersji *Gazety Wyborczej*).

Świat zaczyna korzystać z dysków optycznych do zapisywania wielkich ilości informacji. Na naszym rynku co najmniej od roku są dostępne urządzenia do odczytywania dysków CD ROM, o zasadzie zapisu identycznej, jak w gramofonach kompaktowych. Obecnie firma Archimedes oferuje systemy korzystające z dysków WORM (write once, read many, czyli dyski o jednokrotnym zapisie). System, zawierający także skaner, pozwala na wprowadzenie i zapamiętanie obrazu dokumentu na dysku optycznym. Obraz taki nie może być wymazany. System pozwala równocześnie na dostęp do obrazów dokumentów zapisanych na mikrofilmach i mikrofilmach, ułatwiając pracę np. w sytuacji, gdy stare archiwum było mikrofilmowane, a nowe jest zapisywane na dysku.

Pokazywano także urządzenia, w których komputery pełnią jedynie rolę pomocniczą. Firma Emax demonstrowała zestawy do adresowania, wypisywania czeków i innych podobnych niewielkich nadruków. Urządzenie takie pozwala zastąpić powolną maszynę do odbijania adresów z metalowych matryc (wraz z setkami czy tysiącami matryc, łatwych do pomieszczenia, zagubienia itp.).

dokończenie na s. 27

Współczesne pamięci masowe komputerów osobistych

Analizując fascynujący rozwój konstrukcji komputerów osobistych na przestrzeni ostatniego dziesięciolecia można stwierdzić, że najbardziej spektakularny postęp techniczny skoncentrował się na module pamięci masowej, a więc na tym składniku konfiguracji, który skutecznie blokował możliwość stosowania tej kategorii komputerów do rozwiązywania bardziej złożonych problemów. Jeszcze niedawno wydawało się, że dysk twardy o fantastycznej wówczas pojemności 40 MB stanowi nieprzekraczalną już granicę możliwości technicznych. Dziś do komputera osobistego można przyłączyć jednostki dyskowe o pojemności przekraczającej nawet 1 GB (miliard bajtów!), oferowane przez kilkunastu producentów. Pojawiły się ponadto seryjnie produkowane pamięci laserowe różnych rodzajów, które stwarzają realne i stosunkowo niedoległe perspektywy dalszego zwiększania pojemności. W ten sposób dzisiejszy użytkownik komputera osobistego uzyskuje realne możliwości dysponowania mocą obliczeniową, która jeszcze niedawno była osiągalna jedynie w kategorii komputerów co najmniej średniej wielkości.

Najbardziej zmiennym zjawiskiem są szokująco niskie ceny tych superpamięci, dostosowane do poziomu całkowicie akceptowalnego w tej kategorii sprzętu komputerowego. Spowodowało to drastyczną obniżkę kosztu zapamiętywania jednostkowej informacji, a w konsekwencji znaczną poprawę efektywności przetwarzania danych. Inny czynnik, który może szokować w zestawieniu z dotychczasową praktyką eksploatacji tradycyjnych pamięci masowych to wielokrotne wydłużenie czasu międzyawaryjnego (MTBF), który w odniesieniu do tych wyrobów jest określany w dziesiątkach tysięcy godzin. Ze względu na praktycznie możliwy okres eksploatacji pamięci masowej, tego rodzaju wielkość czasu międzyawaryjnego oznacza jej absolutną niezawodność.

Interesujący przegląd dostępnych na rynku światowym pamięci masowych o pojemności powyżej 500 MB zamieściło w tegorocznym numerze kwietniowym zachodniemieckie czasopismo informatyczne „Personal Computer”. Przegląd ten obejmuje wszystkie cztery rodzaje stosowanych obecnie rozwiązań technologicznych, jakie reprezentują pamięci:

- na dyskach twardych (ang. *hard disk*) z zapisem magnetycznym,
- stałe na dyskach (płytkach) kompaktowych (ang. CD-ROM – *Compact Disk-Read Only Memory*),
- laserowe do wielokrotnego odczytu (ang. WORM – *Write Once, Read Multiple*),
- optyczne z możliwością wielokrotnego zapisu i odczytu informacji (analogicznie jak w pamięci magnetycznej).

Wybrane, najbardziej interesujące parametry pamięci zawartych we wspomnianym prze-

Tabela 1. Pamięci masowe na dyskach twardych

Nazwa wyrobu	Pojemność (MB)	Średni czas dostępu (ms)	Szybkość przesyłania	Czas międzyawaryjny – MTBF (tys. godz.)	Pobór mocy (W)	Okres gwarancyjny (mies.)	Cena (DM)
Control Data CS Disk	500	16	14/18 Mbit/s	40	25	12	9 380
Control Data CS Disk 590	590	16	12/16 Mbit/s	40	27	12	9 580
Fuitsu CS Disk 650	650	16	12/16 Mbit/s	40	27	12	10 650
Fuitsu CS Disk 1020	1000	16	15/23 Mbit/s	40	26	12	18 700
Hitachi DK 515C-78	660	16	1,5/4 MB/s	40	31,5	6	brak danych
Hitachi DK 515-78	780	16	2,48 MB/s	40	30	6	brak danych
HP 97549T	1180	15	5 MB/s	150	34	60	13 500
Maxtor 8760E	768	18	15 Mbit/s	50	27	12	9 270
Micropolis 1518-15	1200	14	4 MB/s	50	31	12	brak danych
Micropolis 1568-15	760	16	15 Mbit/s	150	31	12	7 900
PCST Exfile 760	760	16	15 Mbit/s	40	28	12	10 100
PCST Exfile	1200	16	15 Mbit/s	40	30	12	20 100
Storage LAN650S	650	16	15 Mbit/s	40	27	12	10 400
Storage LAN2-650S	650	16	15 Mbit/s	40	54	12	12 800
WREN V700C	700	14,5	1,3 MB/s	40	27	10	brak danych
Maxtor 8760/S	768	18	4,8 MB/s	50	27	12	10 700

Tabela 3. Pamięci masowe typu WORM (Write Once, Read Multiple)

Nazwa wyrobu	Pojemność (MB)	Średni czas dostępu (ms)	Szybkość przesyłania	Czas międzyawaryjny – MTBF (tys. godz.)	Pobór mocy (W)	Okres gwarancyjny (mies.)	Cena (DM)
Corel CS WORM	940	90	3,8/6,9 Mbit/s	25	30	6	15 000
Maxtor LS 800 AT-I/E	800	108	2,5 Mbit/s	30	brak danych	6	10 000 12 000
Mitsubishi MW-501M11	600	85	1,5 MB/s	20	45	6	7 630
OD-810	810	175	2,18 Mbit/s	brak danych	brak danych	6	15 400
Optimen Serie 600	650	60	725 KB/s	15	45	6	7 700
Optimen 2400/4000	2400	150	766 KB/s	12	45	15	40 160
Ricoh RO 5041	760	108	163 KB/s	20	30	6	brak danych
Ricoh RS 1F	760	168	1 MB/s	20	55	6	brak danych

gładzie, zostały ujęte w zamieszczonych czterech tabelach. Dla każdego z wyrobów podano następujące podstawowe parametry techniczne i eksploatacyjne:

- pojemność (MB),
- średni czas dostępu do informacji (ms),
- szybkość przesyłania (Mb, MB, Kb, lub KB),
- czas międzyawaryjny – MTBF (w tysiącach godzin),
- moc zainstalowana (W),
- udzielany przez producenta okres gwarancyjny (w miesiącach).

W tabelach podano również aktualne ceny tych pamięci (w markach RFN). Są to orientacyjne ceny ofertowe, od których sprzedawcy

stosują – w wyniku coraz ostrzejszej walki konkurencyjnej – upusty w wysokości od 20 do 30%.

Znakomitą większość oferowanych pamięci dostosowano do wstawienia w miejsce używanego dotąd dysku twardego. Jedynie nieliczne modele, których nie udało się zmieścić w gabarytach stosowanych napędów dyskowych, wymagają stosowania odrębnej obudowy. Znaczna część wyrobów dostosowanych do wbudowania w mikrokomputer, może być jednak dostarczana także w odrębnej obudowie. Jest to istotne w przypadku, gdy użytkownik realizuje rozbudowę konfiguracji komputera z zachowaniem już posiadanego dysku twardego.

Wśród producentów wyrobów uwidoczniowych w tabelach wyraźnie dominują firmy japońskie, natomiast z tak licznej niegdys grupy producentów pamięci masowych poprzednich generacji pozostały właściwie dwie firmy: Control Data oraz Hewlett-Packard. Dokumentuje to w szczególności dobitny sposób aktualną sytuację na rynku światowym, gdzie wiele renomowanych firm amerykańskich wskutek niedostatecznego finansowania prac badawczo-rozwojowych musiało ustąpić przed ekspansją nowych, często niewielkich i dotąd całkowicie nieznanymi, przedsiębiorstw azjatyckich.

Należy zauważyć, że wyroby oparte na nowej technologii, chociaż pod względem pojemności całkowicie dorównują, a w niektórych przypadkach nawet przewyższają w tym parametrze pamięci na dyskach twardej, to jednak zdecydowanie ustępują pod względem innych, równie podstawowych parametrów, jakimi są średni czas dostępu do informacji oraz szybkość przesyłania. Nowe technologie wykazują również mniejsze wartości czasu międzyawaryjnego oraz krótsze okresy gwarancyjne, co jest w pełni zrozumiałe z uwagi na wielokrotnie krótszy w tej dziedzinie okres doświadczeń eksploatacyjnych oraz możliwości wprowadzenia ulepszeń konstrukcyjno-technologicznych, czy wreszcie opanowania właściwych metod wytwarzania i kontroli jakości. (WK)

Tabela 2. Pamięci masowe na dyskach kompaktowych (CD-ROM – Compact Disk – Read Only Memory)

Nazwa wyrobu	Pojemność (MB)	Średni czas dostępu (ms)	Szybkość przesyłania	Czas międzyawaryjny – MTBF (tys. godz.)	Pobór mocy (W)	Okres gwarancyjny (mies.)	Cena (DM)
Apple CD-SC	650	55/130	800 KB/s	brak danych	40	12	3000
Data General 6522	600	400	153,6 KB/s	brak danych	8,5	12	1850
Hitachi CDR 1503/S	brak danych	brak danych	brak danych	25	8/15	6	2280
NEC CDR-35	540	1500	150 KB/s	brak danych	brak danych	12	1480
NEC CDR-82	540	350	150 KB/s	20	brak danych	6	1820
NEC CDR-72	540	350	150 KB/s	20	brak danych	6	2050
Philips CDS 201	650	400	176,4 KB/s	brak danych	brak danych	12	1940
Philips CDS 121	650	400	176,4 KB/s	brak danych	brak danych	12	2170
Sanyo ROM 3000 GSI	brak danych	brak danych	brak danych	25	8/15	6	2280
TA CS 500	550	800	153 KB/s	brak danych	brak danych	12	2320
Toshiba CS-ROM Kit 2100	680	350	153,6 Kbit/s	25	8,5	12	brak danych
Toshiba CS-ROM Kit 2200	680	350	153,6 Kbit/s	25	8,5	12	brak danych
Toshiba XM 3201B	680	350	153,6 Kbit/s	25	8,5	12	brak danych

Tabela 4. Pamięci masowe optyczne z wielokrotnym zapisem

Nazwa wyrobu	Pojemność (MB)	Średni czas dostępu (ms)	Szybkość przesyłania	Czas międzyawaryjny – MTBF (tys. godz.)	Pobór mocy (W)	Okres gwarancyjny (mies.)	Cena (DM)
MEO	brak danych	65	680 KB/s	brak danych	brak danych	12	14860
Procom ME-OD 650	650	95	7,4 MB/s	brak danych	23	12	15750
Kontron MOD 650	650	185	620 KB/s	brak danych	brak danych	6	14760
Storage Dimensions	1000	35	10 MB/s	30	50	12	17490
Sumo RSSM600	600	63	12 MB/s	20	33	12	od 14800

INFOSYSTEM '90

dokończenie ze s. 25

Nadal jest rozwijany jedyny chyba naprawdę polski edytor tekstowy TAG. Jego nowa wersja ma duży słownik ortograficzny z opisem podziału na sylaby (do automatycznego przenoszenia słów między wierszami), pozwala na tzw. opracowywanie pomysłów (można łatwo tworzyć schemat tekstu, później dopiero wypełniając go treścią), obsługuje mysz, umożliwia łączenie treści dokumentu (np. listu) z kolejnymi adresami z listy. Producent (spółdzielnia Infoservice) zapowiada wprowadzenie na rynek kolejnego produktu – programu do elektronicznego składu zbliżonego w swoich funkcjach do Pagemakera. Czekamy z niecierpliwością!

HANNA WŁODARSKA
JAROSŁAW DEMINET

Rynek pamięci operacyjnych

dokończenie ze s. 29

o pojemności 4 Mb to firma Oki (układy w konfiguracji $\times 1$ i $\times 4$ w cenie 75 dolarów przy dostawach większych od 10 tys. sztuk), a także tzw. liniowe moduły pamięciowe – SIMM 4×9 po 700 dolarów (przy dostawie od tysiąca sztuk) oraz firma Ramtron (szybkie pamięci o czasie dostępu 60 ns, budowa fabryki w Tateyama o wydajności 1,75 mln kostek miesięcznie).

J.R.

WYDAWNICTWA NAUKOWO-TECHNICZNE

Maksymilian Dryja, Janina i Michał Jankowscy: Przegląd metod i algorytmów numerycznych cz. 2. Warszawa 1988, wydanie 2, nakład 4000 egz., ISBN 83-204-1039-8 całość, ISBN 83-204-1041-X cz. 2

Książka stanowi drugą część monografii poświęconej przeglądowi metod numerycznych. Omówiono w niej podstawowe metody i algorytmy rozwiązywania układów algebraicznych równań liniowych, odwracania macierzy, rozwiązywania równań i układów równań nieliniowych, wyznaczania wartości i wektorów własnych, rozwiązywania liniowego zadania najmniejszych kwadratów oraz zagadnień różniczkowych cząstkowych. Dużo uwagi poświęcono ocenie numerycznej stabilności algorytmów i ich efektywności.

Książka jest przeznaczona dla programistów, pracowników nauki zajmujących się informatyką oraz dla studentów kierunków informatycznych.

Wolfgang Reisig: Sieci Petriego – Wprowadzenie. Warszawa 1988, wydanie 1, nakład 3000 egz., ISBN 83-204-1024-X

Tematem książki są sieci Petriego, czyli pewien formalny sposób opisu współdziałających rzeczywistych systemów, służących do analizy zachodzących w nich zjawisk i znajdujący zastosowanie we wszystkich dziedzinach nauki i techniki. W książce zamieszczono liczne przykłady graficznego przedstawiania zbioru zdarzeń powiązanych relacjami przyczynowości. Omówiono systemy warunków i zdarzeń, sieci miejsc i tranzycji oraz sieci z indywidualnymi znacznikami. Czytelnik znajdzie też ponad sto rysunków sieci oraz zestawy ćwiczeń służących do sprawdzenia stopnia opanowania materiału.

Książka jest przeznaczona dla programistów, pracowników naukowych zajmujących się informatyką, studentów kierunków informatycznych a także dla inżynierów różnych specjalności i organizatorów produkcji.

Ludwig Classen, Ulrich Oefler: Programowanie systemów mikrokomputerowych. Warszawa 1989, wydanie 1, nakład 8000 egz., ISBN 83-204-1125-4

Książka zawiera opis systemów mikroprocesorowych U880, U8000 oraz U881/U882. Wiele uwagi poświęcono zastosowaniom programowalnych układów scalonych oraz zagadnieniom programowania w języku PLZ/ASM i Basic.

Książka jest adresowana do szerokiego kręgu odbiorców, przede wszystkim do programistów, pracowników ośrodków obliczeniowych,

inżynierów zatrudnionych w placówkach badawczo-rozwojowych, studentów i nauczycieli.

Robert Kowalski: Logika w rozwiązywaniu zadań. Warszawa 1989, wydanie 1, nakład 6000 egz., ISBN 83-204-1097-5

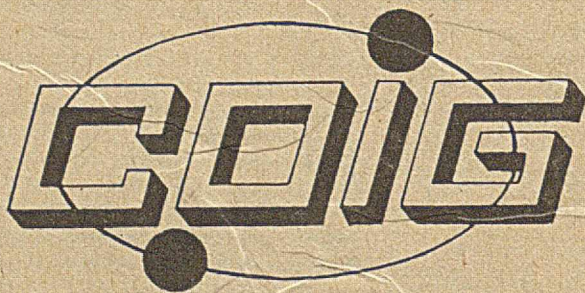
Tematem książki są podstawy teoretyczne programowania w języku logiki. Autor – jeden z twórców tej koncepcji programowania – kładzie główny nacisk na zastosowanie metod logiki w rozwiązywaniu zadań i w bazach danych. Posługuje się przy tym językiem klauzul. Jest to odmiana sformalizowanego języka logiki, wyróżniająca się prostą składnią oraz pozwalająca łatwo zautomatyzować dowodzenie. Na logice klauzul opiera się coraz bardziej popularny język programowania Prolog. W książce znajduje się omówienie różnych metod wnioskowania oraz procedur dowodowych przydatnych w zastosowaniach komputerowych, m.in. wnioskowania wstępującego i zstępującego, reguły rezolucji i procedury grafu połączeń. Oddzielne rozdziały poświęcono zależnościom między językiem przedmiotowym a metajęzykiem oraz ogólnym rozważaniem procesów zachodzących w systemie zmieniającej się informacji.

Zagadnienia omówione w książce nie ulegają dezaktualizacji. Autor unika nadmiernego formalizowania wywodów, dzięki czemu książka nie wymaga od Czytelnika przygotowania matematycznego ani informatycznego, zmusza jednak do wysiłku i zaangażowania intelektualnego. Jest przeznaczona dla programistów, projektantów systemów przetwarzania informacji i systemów doradczych, pracowników nauki zajmujących się informatyką oraz dla studentów kierunków matematycznych i informatycznych.

JClaude Delobel, Michel Adiba: Relacyjne bazy danych. Warszawa 1989, wydanie 1, nakład 5000 egz., ISBN 83-204-1025-8

W książce przedstawiono problematykę baz danych zarówno od strony praktycznej, jak i teoretycznej. Przy opisie baz danych skoncentrowano się na podejściu relacyjnym, takie podejście bowiem pozwala najlepiej zrozumieć i rozwiązać podstawowe problemy związane z tworzeniem baz danych dla potrzeb rozwiązywania problemów praktycznych. Książka składa się z trzech części – w części pierwszej wprowadzono podstawowe pojęcia i funkcje systemu baz danych oraz omówiono modele baz danych hierarchiczne, sieciowe i relacyjne; w drugiej części omówiono zagadnienia związane z oprogramowaniem baz danych, charakterystyczne dla podejścia relacyjnego; część trzecia jest poświęcona zagadnieniom teoretycznym procesu tworzenia schematów relacyjnych baz danych.

Książka jest przeznaczona dla użytkowników i projektantów baz danych oraz dla studentów kierunków informatycznych.



CENTRALNY OŚRODEK INFORMATYKI GÓRNICZWA
40-065 Katowice, ul. Mikołowska 100
tel. 574-777, telex 00313711, telefaks 517-442

**PROJEKTOWANIE I PROGRAMOWANIE SYSTEMÓW INFORMATYCZNYCH.
KOMPLETACJA, SPRZEDAŻ SYSTEMÓW I SPRZĘTU KOMPUTEROWEGO, KSEROKOPIAREK,
TELEFAKSÓW, PODZESPOŁÓW I CZĘŚCI ZAMIENNYCH.
INSTALACJA, NAPRAWY I PRZEGLĄDY URZĄDZEŃ MIKROKOMPUTEROWYCH.
SZKOLENIE PROGRAMISTÓW, PROJEKTANTÓW I OPERATORÓW SPRZĘTU INFORMATYCZNEGO.**

Oferujemy usługi w zakresie ujmowania i przetwarzania danych w trybie wsadowo-partiowym oraz zdalnym realizowane na komputerach: MERA-9150, ODRA-1300, ICL-1900, 2900, 39, RIAD.

Sprzęt mikrokomputerowy kompatybilny z IBM PC konfigurujemy zgodnie z potrzebami użytkownika.

Instalujemy sieci lokalne. Tworzymy systemy aplikacyjne.

Prowadzimy serwis sprzętu mikrokomputerowego w zakresie instalacji, napraw i przeglądów.

Dostarczamy wymagane oprogramowanie narzędziowe.

Szkolimy z zakresu obsługi mikrokomputerów, systemów operacyjnych, sieci lokalnych i teletransmisji, języków programowania oraz generatorów baz danych.

Projektujemy systemy informatyczne wg założeń odbiorcy lub na podstawie analizy jego potrzeb.

Dostarczamy oprogramowanie gotowych projektów z zastosowaniem dowolnego języka.

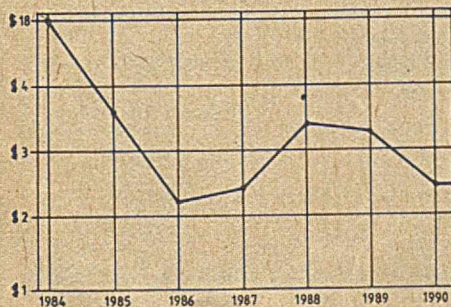
EO/943/89

Rynek pamięci operacyjnych

Pamięci półprzewodnikowe pojawiły się na rynku w 1970 r., kiedy firma Intel wprowadziła dynamiczne pamięci operacyjne (DRAM – *Dynamic Random Access Memory*) oznaczone symbolem 1108 o pojemności 1 Kb w cenie 35 dolarów. Wkrótce też dali o sobie znać konkurenci z Japonii, którzy w 1973 roku wyprodukowali 8 milionów tych układów, a w rok później wprowadzili układy o pojemności 4 Kb. W latach 1978–1979 w Japonii i USA równocześnie rozpoczęto realizację programów produkcji bardzo szybkich układów scalonych. W roku 1980 pojawiły się w Japonii kostki o pojemności 64 Kb, a rok później w USA już jedenaście firm produkowało układy DRAM (AMD, AMI, Fairchild, Intel, ITT, Micron, Mostek, Motorola, National Semiconductor, Texas Instruments i Zilog). W roku 1984 do współzawodnictwa włączyły się firmy zachodnioeuropejskie Siemens i Philips, które rozpoczęły realizację programu Mega, mającego doprowadzić w 1989 r. do opracowania układów o pojemności 4 Mb.

W latach 1985–1986 japońscy producenci DRAM tak znacznie obniżyli ceny swych wyrobów, że w USA potraktowano je jako ceny dumpingowe. Doprowadziło to z jednej strony do zaprzestania produkcji tych układów przez wspomnianych producentów amerykańskich, z wyjątkiem dwóch firm – Micron i Texas Instruments, z drugiej zaś do podpisania w 1986 r. porozumienia amerykańsko-japońskiego ograniczającego nadmierny import do USA i ustalającego minimalne ceny.

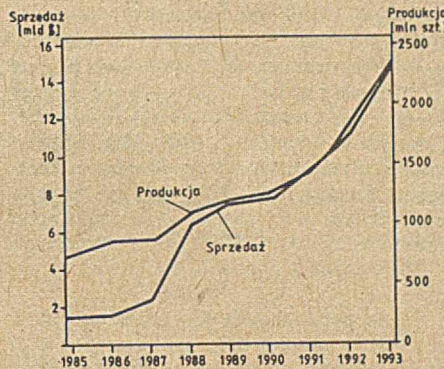
Łącznie ze zwiększonym zapotrzebowaniem na układy pamięciowe spowodowało to w latach 1987–1988 wystąpienie niedoboru tych układów na rynku, co z kolei spowodowało ponowny wzrost cen.



Rys. 1. Zmiana ceny kostki DRAM o pojemności 256 KB w latach 1984–1990

Szczególnie wyraźną tendencję wzrostu cen wykazały układy DRAM o pojemności 256 Kb. (rys. 1) Tendencję tę potęgował fakt, że w oczekiwaniu na układy o pojemności 1 Mb, których pełna produkcja opóźniła się, nie podejmowano wytwarzania układów o mniejszej pojemności.

Analiza kształtowania się produkcji i sprzedaży układów DRAM w okresie lat 1985–1993 wykazała, że po zastoiu produkcji w latach 1986–1987 nastąpił jej wzrost w okresie 1987–1988, następnie znów niewielkie przyhamowanie w latach 1988–1989 i większe w okresie 1989–1990, po którym nastąpi przyspieszony wzrost do poziomu około 2,3 mld elementów w 1993 roku (rys. 2). Rynek po gwałtownym wzroście obrotów w latach 1987–1988 (z 2 do ponad 6 mld dolarów) wykazał w latach 1989–1990 zmniejszony ich wzrost i stabilizację na poziomie 7 mld dolarów. Przewiduje się, że od 1990 r. obroty będą wzrastać coraz szybciej aż do ok. 15 mld w roku 1993.



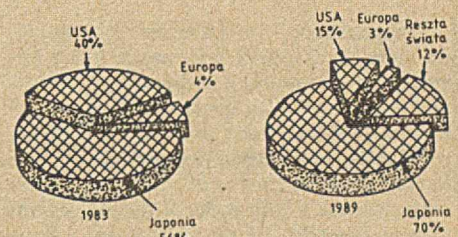
Rys. 2. Produkcja i sprzedaż elementów DRAM w latach 1985–1993

Procentowy udział w tej produkcji poszczególnych regionów świata uległ w 1989 r. w porównaniu z 1983 r. znacznym zmianom (rys. 3).

Najbardziej zaskakujące prawie trzykrotne zmniejszenie się udziału USA. Jeśli chodzi o wykorzystanie elementów pamięciowych, to w 1988 r. 53% układów DRAM przeznaczono do budowy komputerów osobistych i automatyzacji prac biurowych, 24% do budowy innych komputerów, 8,8% dla telekomunikacji, 6,6% do celów przemysłowych i innych, 5% dla urządzeń elektroniki użytkowej oraz 2,6% do urządzeń wojskowych. Przewidywania na rok 1992 (przy wzroście wartości tych elementów o 43% w stosunku do roku 1988) mówią o pewnym zmniejszeniu udziału wykorzystania w komputerach (łącznie z 77 do 70,5%), natomiast o wzroście w telekomunikacji do 10,3%, urządzeniach elektroniki użytkowej do 8,6% i wojskowych do 3,5%.

Podobne dane dla pamięci statycznych SRAM (*Static Random Access Memory*), których sprzedaż była w 1988 r. prawie czterokrotnie mniejsza niż pamięć DRAM (1,7 mld dolarów w porównaniu do 6,3 mld) mówią o przeznaczaniu 35,9% do budowy komputerów osobistych i automatyzacji prac biurowych, 22,8% do innych komputerów, 16,9%

do zastosowań przemysłowych i innych, 10,7% – wojskowych; 8,8% – telekomunikacyjnych i 5% do urządzeń elektroniki użytkowej. Proporcje te mają być mniej więcej zachowane w 1992 r. przy pewnym wzroście w działach komputerów osobistych, zastosowań wojskowych i urządzeń elektroniki użytkowej, natomiast zmniejszeniu w pozostałych dziedzinach, przy czym wzrost wartości sprzedanych elementów będzie tu większy (54,5%) niż dla pamięci DRAM (43,3%).



Rys. 3. Udział różnych regionów świata w rynku DRAM w 1983 i 1989.

Udział w produkcji światowej (%)

	1983	1989
Japonia	56	70
USA	40	15
Europa	4	3
Pozostałe kraje	-	12

Pierwotnie szacowano, że w 1989 r. zostanie sprzedanych 700 mln kostek DRAM o pojemności 256 Kb oraz 500 mln o pojemności 1 Mb, natomiast w roku 1990 proporcja ta miała się odwrócić: 700 mln 1 Mb i 500 mln 256 Kb. We wrześniu 1989 roku, gdy pojawiły się pierwsze oferty kostek o pojemności 4 Mb, ceny kostek 1 Mb spadły z 15 do 11–12 dolarów. Analitycy z firmy Dataquest przewidywali obniżkę tych cen w końcu 1989 r. do 10 dolarów oraz do 9 dolarów w roku 1990. Inne prognozy mówią nawet o cenie 7–8 dolarów pod koniec 1990 roku.

W październiku 1989 roku podano, że japońskie firmy Toshiba i Hitachi opóźnią o pół roku (od marca do września 1990 r.) rozpoczęcie pełnej produkcji układów o pojemności 4 Mb, a jednocześnie podano o podjęciu współpracy w tym zakresie między IBM z jednej strony, a takimi firmami, jak: Micron, US Memories i Cypress Semiconductor z drugiej. Nieco bardziej opóźniona jest tu firma Motorola, która dopiero w pierwszej połowie 1991 roku ma oferować takie elementy wyprodukowane we współpracy z Toshiba, w cenie 110 dolarów przy układach o czasie dostępu 100 ns oraz 125 dolarów przy układach 80 ns. Inni potencjalni wytwórcy kostek

dokończenie na s. 27

Drukarka laserowa firmy Hewlett-Packard

Zwykle usprawnienie sprzedawanego wyrobu łączy się z niewielką podwyżką dotychczasowej ceny, bądź podkreśla się jej utrzymanie bez zmian, mimo większych walorów nowego modelu. Oferta nowej drukarki laserowej Laser-Jet III jest jeszcze korzystniejsza. Przy znacznych ulepszeniach w stosunku do będącej niemal standardem światowym Laser-Jet II, Hewlett-Packard ustalił cenę nowej drukarki na 2395 dolarów, a więc o 300 dolarów niższą od poprzedniego wyrobu.

Przypomnijmy, że Laser-Jet II należała do tańszych drukarek laserowych, które powszechnie stosowano z komputerami osobistymi. Drukarka ta stwarzała możliwość dysponowania 3-4 zestawami czcionek o ustalonych wymiarach. Inną alternatywą była droga drukarka z językiem PostScript oferująca 35 zestawów czcionek o zmiennych wymiarach. Tymczasem Laser-Jet III nie korzystając z PostScriptu daje identyczne możliwości, które w dodatku są nadal rozszerzane. Na czerwiec 1990 r. zapowiedziano dysk ze 108 zestawami czcionek o zmiennych wymiarach w cenie zaledwie 199 dolarów.

Nowa drukarka zachowuje pełną prędkość mechanizmu drukującego – 8 stron na minutę, podczas gdy „ekonomiczna” wersja Laser-Jet II (w cenie 1495 dolarów) drukowała z prędkością dwukrotnie mniejszą. Również złożone pod względem wydawniczym strony zawierające grafikę i inne efekty specjalne, drukowane są o 46% szybciej dzięki zastosowaniu szybszego procesora, lepszemu algorytmu buforującego i zawartego w pamięci stałej nowego języka sterowania PCL5. Lepsza jest też jakość wydruku. Uzyskuje się tu dalszą poprawę rozdzielczości poprzez możliwość 16-krotnej zmiany wymiaru elementu świetlnego i jego położenia w ramach standardowego kwadratu o boku 1/300 cala. Realizacja następuje przez modulację szerokości impulsów laserowych. Dotychczas stosowano takie rozwiązanie tylko w drogich drukarkach.

Drukarka ma podobne wymiary jak Laser-Jet II (22 x 46 x 64 cm) i waży 23 kg. Model standardowy wyposażony jest w 1 MB pamięci operacyjnej, z czego 720 KB jest zarezerwowane na buforowanie strony (w Laser-Jet II tylko 395 KB). Pamięć można rozszerzyć aż do pojemności 5 MB. Jeśli chodzi o pojemniki tonera, kasety czcionek itp. elementy wyposażenia, to nowa drukarka jest całkowicie zgodna z modelem Laser-Jet II.

Producent przygotowuje jednocześnie rozwiązanie o nazwie Formatter, które pozwoli na uzyskanie w Laser-Jet II niektórych parametrów eksploatacyjnych nowej drukarki. Przewiduje się, że wspomniane rozwiązanie pojawi się na rynku w czerwcu 1990 r. i będzie kosztowało ok. 1100 dolarów. (JR)

Stowarzyszenie użytkowników informatyki w NRD

Mimo znacznie większego niż u nas rozwoju informatyki, dopiero 24 stycznia br. rozpoczęło działalność Stowarzyszenie Niemieckich Użytkowników Informatyki w NRD (ADI-DDR – *Anwenderverband Deutscher Informationsverarbeiter/DDR*). Stowarzyszenie to jest odpowiednikiem istniejącej już w RFN organizacji o analogicznej nazwie ADI-BRD (ADI-RFN) i będzie przez nią finansowo wspomagane.

ADI-DDR jest wspólną organizacją zawodową praktyków i naukowców. Podstawowym zadaniem Stowarzyszenia jest popieranie racjonalnego i efektywnego zastosowania informatyki w gospodarce, administracji publicznej i nauce oraz ochrona interesów zawodo-

wych członków. Punktem ciężkości w działalności stowarzyszenia jest szeroko pojęta wymiana doświadczeń między praktykami i naukowcami. ADI-DDR ubiega się obecnie o członkostwo Europejskiego Stowarzyszenia Informatyków CECUA, mającego siedzibę w Brukseli. Członkami Stowarzyszenia mogą być osoby zarówno fizyczne, jak i prawne.

26 maja br. odbyło się uroczyste zebranie założycielskie, podczas którego ustalono, że pierwsza ogólnoniemiecka konferencja użytkowników odbędzie się w dniach 21–22 września br. w Berlinie. Będzie ona miała charakter seminarium adresowanego do kadr kierowniczych. Adres ADI-DDR: ADI-Orgbüro, PSF 22, Karl-Marx-Stadt, 9025.

Współpraca SIEMENSA z IBM

Koncerny SIEMENS i IBM zawarły ostatnio porozumienie dotyczące współpracy w zaprojektowaniu pamięci typu DRAM (*Dynamic Random Access Memory*) o pojemności 64 Mbitów. Porozumienie to zawiera również stwierdzenie o możliwości przyszłego rozszerzenia współpracy na nowe generacje układów scalonych. Udział obu partnerów w ponieszeniu kosztów prac tego przedsięwzięcia ustalono w równych częściach. Prace zostaną podjęte natychmiast i będą koncentrować się głównie w nowo utworzonym centralnym ośrodku badawczym nowoczesnych technologii półprzewodnikowych (*Advanced Semiconductor Technology Center*) firmy IBM w East Fishkill, N. Y. (USA) oraz opierać się na działalności badawczej ośrodków Siemens w Monachium i IBM w Essex Junction, Vt. (USA).

Celem porozumienia jest uruchomienie w połowie lat dziewięćdziesiątych seryjnej produkcji wspomnianego układu pamięci o pojemności 64 Mb. Chociaż partnerzy nie ujawnili kosztów tego przedsięwzięcia, niezależni spec-

jaliści szacują wielkość niezbędnych nakładów na ok. 450 mln dolarów USA. Wspólne badania będą wykorzystywać m.in. to, że IBM stosuje już osmiocalowy płat krzemowy, który w porównaniu do stosowanego dziś płatu sześciocalowego ma powierzchnię większą o prawie 80%. Umożliwią to wytworzenie z jednego platu odpowiednio większej liczby kostek pamięci, a w konsekwencji – i w istotnym stopniu obniżenie kosztów ich wytwarzania.

Dokładna pojemność projektowanego układu pamięci wyniesie 67108864 bitów, co odpowiada pojemności informacyjnej ponad 3000 znormalizowanych stron maszynopisu. Przy okazji należy przypomnieć, że obie firmy w 1989 r. uruchomiły produkcję seryjną układów o pojemności 4 Mbitów, a od pewnego czasu prowadzą niezależnie intensywne prace badawcze nad następną generacją układów pamięciowych o pojemności 16 Mbitów. Obecne nawiązanie współpracy należy rozumieć jako odpowiedź na coraz groźniejszą w tej dziedzinie konkurencję japońską.

Konferencja CASE

BCS CASE Specialist Group (Seksja CASE Brytyjskiego Towarzystwa Informatycznego) organizuje w dniach 9–11 września 1990 r. w Robinson College Uniwersytetu Cambridge (W. Brytania) konferencję naukową (połączoną z wystawą) pod nazwą „CASE on Trial” („CASE po okresie prób”).

CASE (*Computer Aided Software Engineering*) – Inżynieria Oprogramowania Wspomagana Komputerem – jest nową, bardzo efektywną, ale mało jeszcze rozpowszechnioną technologią tworzenia oprogramowania systemów informatycznych.

Program konferencji obejmuje znaczną liczbę nadesłanych referatów na takie tematy, jak:

- korzyści ze stosowania CASE,
- przyszłościowe wersje CASE,
- rozwój, użytkowanie i ocena narzędzi CASE,
- zastosowanie CASE,
- problemy organizacji i zarządzania w CASE,
- posługiwanie się CASE.

Główną atrakcją Konferencji będą trzy referaty autorstwa czołowych światowych ekspertów technologii CASE:

- David Fairbairn (W. Brytania): *The Opportunity of the Decade* (Szansa obecnego dziesięciolecia),
- Michael Jackson (W. Brytania): *CASE Tools and Development Methods* (Narzędzia CASE oraz metody ich wykorzystania),
- Paul Ward (USA): *The Use of Traditional CASE Tools in Object Oriented Analysis and Design* (Użycie tradycyjnych narzędzi CASE w obiektowej analizie i projektowaniu).

Szczegółowych informacji na temat programu i uczestnictwa w Konferencji udziela Jenny Oakey, BISL Conferences, 703/4 Delta Business Park, Welton Road, SWINDON SN5 7XS, England. Tel.: +44 793 480269, telefaks: +44 793 480270.

Docierają do redakcji sygnały z całego kraju o trudnościach z nabyciem INFORMATYKI w kioskach Ruchu.

Pragniemy wyjaśnić Czytelnikom przyczynę tych kłopotów. Redakcja podjęła decyzję zmniejszenia liczby egzemplarzy skierowanych do sprzedaży kioskowej, gdyż Ruch pobiera bardzo wysoką prowizję za swoją usługę, nie gwarantując prawidłowego rozdziału egzemplarzy w sieci kioskowej.

W związku z tym prosimy wszystkich zainteresowanych nabyciem INFORMATYKI o kontakt z redakcją (tel. 39-14-34) lub Działem Handlowym SIGMY (tel. 40-30-86). – będziemy Państwu wysyłać egzemplarze za zaliczeniem pocztowym – oczywiście w miarę naszych możliwości dość ograniczonych wysokością nakładu. Czytelnicy z terenu i okolic Warszawy mogą kupić poszczególne numery w Klubie SIGMY przy ul. Mazowieckiej 12, w księgarni „Domu Książki” przy ul. Mokotowskiej 51/53 oraz w lokalu redakcji.

Jednocześnie przypominamy, że najpewniejszą formą otrzymywania INFORMATYKI jest prenumerata, gdyż ona głównie kształtuje wysokość nakładu. Szczegóły jej zamawiania podajemy na stronie II okładki.

**Firma software'owa
z udziałem zagranicznym**

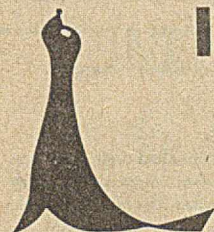
zatrudni informatyków

**mających doświadczenie
w projektowaniu i wykonywaniu
aplikacji
pod MS-WINDOWS
oraz
systemem UNIX
w językach czwartej generacji**

Oferty:

**30-444 Kraków 55
skrytka pocztowa 37**

0/6/90



INTERSOFT

Sp. z o.o.

**Nasze Biuro Handlu Zagranicznego poleca swoje usługi
w zakresie eksportu-importu każdego towaru.
Prowadzi sprzedaż
profesjonalnego sprzętu komputerowego i peryferii.**

**1. JĘZYKI PROGRAMOWANIA,
BIBLIOTEKI, GRAFIKA**

Basic – programowanie w języku Basic	
Cz. III	52 000 zł
Turbo Basic	183 000 zł
Professional Fortran + dodatki	144 000 zł
Fortran 77	132 600 zł
Fortran 80	65 300 zł
Język C – zastosowanie dla zaawansowanych	214 200 zł
Kompilator języka C	106 000 zł
System Turbo C V.1.5	79 000 zł
Turbo C biblioteka	150 000 zł
Turbo C podręcznik użytkownika	109 000 zł
Turbo Pascal	123 500 zł
Turbo Pascal V.4.0	277 000 zł
Turbo Pascal V.5.0	161 000 zł
Turbo Pascal V.5.5 dodatek do T.P.V.5.0	56 000 zł
Turbo Power Tools Plus	124 000 zł
Turbo Database Toolbox	33 000 zł
Moduła 2.86	125 500 zł
Turbo Pascal 5.0 opis języka	189 000 zł
Turbo Pascal 4.0 (aut. Jan Bielecki)	153 000 zł

2. BAZY DANYCH, PAKIETY ZINTEGROWANE

Clipper do dBase III	30 000 zł
Clipper do dBase III +	131 000 zł
Clipper 86	158 000 zł
Clipper 87	212 700 zł
Clipper 87 do dBase III +	214 000 zł
dBase II i III – podręcznik dla zaawansowanych	205 000 zł
dBase III + instalacja pakietu relacyjnej bazy danych	22 000 zł
dBase III + zastosowanie	205 000 zł

dBase III + opis pakietu sieciowego	55 000 zł
dBase III + programowanie	126 000 zł
dBase III + poznanie	111 000 zł
dBase III poradnik encyklopedyczny	237 000 zł
dBase IV	249 000 zł
Framework II P	162 000 zł
Informix	261 000 zł
Lotus 1-2-3 + dodatek (komplet)	113 000 zł
Multiplan	66 000 zł
Pakiet C-ISAM	34 700 zł
CX – Fortran kompilator języka do systemu DOS2	114 300 zł
Plib 86/Plink 86 opis języka	47 000 zł
Statgraphics	91 000 zł
Programowanie systemów baz danych	
języka Clipper	107 000 zł
Kompilator CBasic tom. II	92 000 zł

**3. SYSTEMY OPERACYJNE,
PROGRAMY UŻYTKOWE**

System operacyjny DOS cz. 1 i 2	155 000 zł
System operacyjny DOS 3.10	47 000 zł
System operacyjny DOS 3.20	181 000 zł
System operacyjny DOS 3.30	180 000 zł
System operacyjny DOS 4.0	202 000 zł
Eureka	64 000 zł
Norton Commander	63 000 zł
Polywindows	50 000 zł
Sidekick	44 000 zł

4. EDYTORY

Chl-2 Writer	31 000 zł
Personalny Edytor PE	68 300 zł
Wordstar 2000	142 000 zł

5. CAD-SIECI

AutoCad 2.17	137 000 zł
AutoCad	30 600 zł
Lanlink 4.0 sieć lokalna	80 000 zł
NOWELL, instalacja sieci (Arcnet)	9 000 zł
NOWELL, instalacja sieci	65 000 zł
NOWELL, podręcznik użytkownika sieci	116 000 zł
Or-Cad	134 000 zł

6. ROZMAIŃCOCI

Przewodnik programisty	205 000 zł
Printer STAR NL-10 instrukcja	20 000 zł
Wprowadzenie do użytkowania komputerów osobistych klasy IBM PC	43 000 zł
Drukarka NX-16 instrukcja	106 000 zł
Drukarka	14 000 zł
Karta system rozliczania kart drogowych	10 000 zł
Appagen	18 000 zł

Zamówienia prosimy kierować pod adresem:

**INTERSOFT Sp. z o.o.
00-496 Warszawa, ul. Krucza 16/22,
telefony: 28-44-81 w. 284, 352; 28-29-53,
telex: 812414**

0/7/90

<p>Jurczak L., Warski W.: System baz danych CX-DMOS (I)</p> <p>INFORMATYKA 1990, nr 7, s. 1</p> <p>Pierwsza część charakterystyki wielodostępnego systemu baz danych CX-DMOS, obejmująca ogólne omówienie budowy i zasad działania systemu.</p>	<p>Jurczak L., Warski W.: CX-DMOS database system (I)</p> <p>INFORMATYKA 1990, No. 7, p. 1</p> <p>First part of characteristics of the CX-DMOS multiaccess database system, which includes general discussion of the structure and operation principles of the system.</p>	<p>Jurczak L., Warski W.: CX/DMOS-Datenbanksystem (I)</p> <p>INFORMATYKA 1990, Nr. 7, S. 1</p> <p>Erster Teil einer Charakteristik von CX-DMOS, einem Datenbankmehrbenutzersystem, der eine allgemeine Beschreibung der Struktur, sowie Betriebsgrundlagen des Systems umfasst.</p>
<p>Rybiński H., Sobolewski P.: GT-HOLMES – graficzna baza danych</p> <p>INFORMATYKA 1990, nr 7, s. 4</p> <p>Charakterystyka projektu systemu bazy danych GT-HOLMES stanowiącego rozszerzenie systemu HOLMES w kierunku graficznych baz danych. Omówiono funkcje graficznej bazy danych i język systemu GT-HOLMES oraz podano przykłady korzystania z mechanizmów graficznych w tym języku.</p>	<p>Rybiński H., Sobolewski P.: GT-HOLMES – a graphic database</p> <p>INFORMATYKA 1990, No. 7, p. 4</p> <p>Characteristics of the GT-HOLMES database system design, which is an extension of the HOLMES system in direction to graphic databases. Functions of graphic database and language of the GT-HOLMES system, as well as examples for graphic mechanisms used in the language are discussed.</p>	<p>Rybiński H., Sobolewski P.: GT-HOLMES – eine graphische Datenbank</p> <p>INFORMATYKA 1990, Nr. 7, S. 4</p> <p>Eine Charakteristik des Entwurfes von GT-HOLMES-Datenbanksystem, das eine Erweiterung von HOLMES-System in Richtung der graphischen Datenbank darstellt. Es wurden Funktionen der graphischen Datenbank und die Sprache des Systems, sowie Beispiele der Nutzung von graphischen Mechanismen dieser Sprache, behandelt.</p>
<p>Białas A., Bojda A.: Sieci MAP i TOP. Przegląd standardów i przykład aplikacji</p> <p>INFORMATYKA 1990, nr 7, s. 9</p> <p>Charakterystyka standardów sieci komputerowych MAP i TOP oraz przykład zastosowania tych standardów do sieci NETISS, przeznaczonych do zastosowań przemysłowych i laboratoryjnych z użyciem komputerów klasy MERA 80 oraz IBM PC.</p>	<p>Białas A., Bojda A.: MAP and TOP networks. Standards survey and application example</p> <p>INFORMATYKA 1990, No. 7, p. 9</p> <p>Characteristics of the MAP and TOP computer network standards and an example of their application in the NETISS network, which is appropriated for industrial and laboratorial applications using computers of the MERA 80 and IBM PC category.</p>	<p>Białas A., Bojda A.: MAP und TOP Netze. Normenübersicht und Anwendungsbeispiel</p> <p>INFORMATYKA 1990, Nr. 7, S. 9</p> <p>Eine Charakteristik von Normen für MAP und TOP Netze, sowie ein Beispiel der Anwendung dieser Normen im NETISS-Netz, vorgesehen für Industrie- und Laboranwendungen mit Einsatz der Computer von MERA 80- und IBM PC-Kategorie.</p>
<p>Szczawińska D., Trawiński B.: Konwersja danych między pakietami micro CDS/ISIS z dBase III Plus</p> <p>INFORMATYKA 1990, nr 7, s. 13</p> <p>Charakterystyka konwersji danych, umożliwiającej przeniesienie baz bibliograficznych, zrealizowanych w systemie dBase III Plus, do systemu micro CDS/ISIS, specjalnie zaprojektowanego i bardziej efektywnego w obsłudze tego typu baz.</p>	<p>Szczawińska D., Trawiński B.: Data conversion between Micro CDS/ISIS and dBase III Plus packages</p> <p>INFORMATYKA 1990, No. 7, p. 13</p> <p>Characteristics of data conversion, which enables transfer of bibliographic data bases realized in the dBase III Plus system to the Micro CDS/ISIS, a specially designed and in servicing more effective system for such bases.</p>	<p>Szczawińska D., Trawiński B.: Datenumwandlung zwischen Systemen Micro CDA/ISIS und dBase III Plus</p> <p>INFORMATYKA 1990, Nr. 7, S. 13</p> <p>Eine Charakteristik von Datenumwandlung, die eine Übertragung der mit dem dBase III Plus realisierten bibliographischen Datenbanken zum Micro CDS/ISIS, einem speziell entworfenen und viel mehr effektiven in Bedienung solcher Datenbanken System, ermöglicht.</p>
<p>Wierzbę A.M.: Edytor schematów elektronicznych</p> <p>INFORMATYKA 1990, nr 7, s. 17</p> <p>Charakterystyka opracowanego w Instytucie Informatyki Uniwersytetu Warszawskiego programu rysowania schematów elektronicznych, który w porównaniu do podobnych rozwiązań standardowych zawiera wiele usprawnień. Omówiono kierunki przyszłej rozbudowy programu.</p>	<p>Wierzbę A.M.: Electronic scheme editor</p> <p>INFORMATYKA 1990, No. 7, p. 17</p> <p>Characteristics of the in Warsaw University Informatics Institute elaborated program for electronic scheme drawing, which presents some improvements in comparison to similar standard solutions. Directions of future development of the system are discussed.</p>	<p>Wierzbę A.M.: Editor für elektronische Schemata</p> <p>INFORMATYKA 1990, Nr. 7, S. 17</p> <p>Eine Charakteristik des im Informatik Institut der Warschauer Universität erarbeiteten Programms für Zeichnen der elektronischen Schemata, das im Vergleich mit ähnlichen Standardlösungen viele Verbesserungen enthält. Es wurden Richtungen der künftigen Erweiterung des Programmes besprochen.</p>
<p>Nguyen Kim Sach: Sprzętowe wzmacnianie obrazów cyfrowych w czasie rzeczywistym</p> <p>INFORMATYKA 1990, nr 7, s. 20</p> <p>Możliwości poprawy jakości cyfrowych obrazów telewizyjnych cyfrowych w wyniku zastosowania komputerów o dużej mocy obliczeniowej.</p>	<p>Nguyen Kim Sach: Real-time amplification of digital images by hardware means</p> <p>INFORMATYKA 1990, No. 7, p. 20</p> <p>Possibilities of the TV-digital images quality improvement using high performance computers.</p>	<p>Nguyen Kim Sach: Die hardwareartige Verstärkung von Digitalbildern im Echtzeitverfahren</p> <p>INFORMATYKA 1990, Nr. 7, S. 20</p> <p>Möglichkeiten von Qualitätsverbesserung der Fernseh-digitalbilder infolge der Anwendung von Grossrechnern.</p>
<p>Stożek J.: FidoNet – międzynarodowa amatorska sieć komputerowa</p> <p>INFORMATYKA 1990, nr 7, s. 22</p> <p>Charakterystyka sposobu działania oraz możliwości korzystania z międzynarodowej sieci komputerowej FidoNet.</p>	<p>Stożek J.: FidoNet – an international amateur computer network</p> <p>INFORMATYKA 1990, No. 7, p. 22</p> <p>Characteristics of the FidoNet international network operation principles and access possibilities.</p>	<p>Stożek J.: FidoNet – eine internationale Computernetz</p> <p>INFORMATYKA 1990, Nr. 7, S. 22</p> <p>Eine Charakteristik von Betriebsmethode und Nutzungsmöglichkeiten des internationalen FidoNet-Computernetzes.</p>

Jest lato 1990 roku. Zostaliśmy już odarci ze złudzeń możliwości wygodnego życia na czyjś rachunek. Okazało się, że stosunkowo prosty zabieg ekonomiczny zmienił radykalnie układ zależności w wielu dziedzinach. Coś, co w zeszłym roku trzeba było zalać, obecnie można kupić, tylko trzeba mieć pieniądze. Taki stan dotknął też **INFORMATYKĘ** – jedyne pismo naukowo-techniczne z tej dziedziny w kraju. Niewiele brakowało, aby numer 8–12 z 1989 roku był numerem ostatnim. Na razie Redakcja, mocno zmęczona, wybroniła się, chociaż musiała gwałtownie pobierać lekcje nowej ekonomii.

Oczywiście, tak jak w kapitalizmie, nic już nie jest pewne. To popyt na rynku zadecyduje, kto będzie miał szansę dalszego istnienia. Inaczej mówiąc, od Was Czytelnicy zależy, czy **INFORMATYKA** będzie się pojawiać regularnie na rynku. Jasne jest przy tym, że podstawowym motywem zaakceptowania przez Was nowej, stosunkowo wysokiej ceny pisma (choć jest to tylko sześć przejazdów autobusem pospiesznym!) będzie jakość jego treści, a to zależy głównie od Redakcji. Ale z kolei to nie Redakcja zapełnia łamy pisma swoimi artykułami, ale stara się wybrać najlepsze prace napisane przez najlepszych – czyli przez Was.

W ten sposób zdefiniowałem nieskończoną pętlę, co jak wiemy z praktyki programowania musi się zakończyć zewnętrzną ingerencją. Kto ma to robić? Oczywiście demokratyczne głosowanie nad dalszymi losami pisma przeprowadzicie Wy – informatycy pozostający w kraju w nadziei, że zbliżanie do Europy musi pociągnąć za sobą zbliżenie w technologii komputerowej. A głosowanie to będzie polegało na bardziej aktywnym niż dotychczas zasilaniu nas ciekawymi materiałami oraz popieraniu naszych starań o dalsze istnienie **INFORMATYKI** w Waszym środowisku, co powinno wyrażać się przede wszystkim w kupowaniu, a najlepiej przenie-

rowaniu naszego wspólnego czasopisma. Proszę sobie bowiem wyobrazić stan, gdy przy braku dostępu do literatury zachodniej oraz braku polskiej literatury technicznej (wydawnictwa techniczne są na granicy bankructwa) zabraknie również stosunkowo najszybszej informacji o nowościach i trendach w informatyce światowej.

Tyle o problemach Redakcji. Od siebie chciałbym dodać, że postanowiłem kontynuować współpracę z pismem, traktując to w dużym stopniu, jako działalność społeczną. Żywię nadzieję, że obecna konieczność żywszego zainteresowania się – z przyczyn ekonomicznych – sensem własnej pracy, a nawet groźba utraty tej pracy, pobudzi wielu ludzi do zaprezentowania swoich możliwości. Proszę bowiem wziąć pod uwagę fakt, że w staraniach o atrakcyjną pracę należy zwykle wykazać się swoimi dotychczasowymi osiągnięciami, a nie zawsze kawałek wydruku programu zapisany w jakimś języku programowania jest do tego najlepszą rekomendacją.

Swoje niezależne poglądy na różne tematy dotyczące informatyki postanowiłem przekazać zapisując **ZBITki** myśli i spostrzeżeń. Oczywiście uzyskane od Was Czytelnicy opinie będą miały wpływ również na kształt moich wypowiedzi, a także na sens ich kontynuacji na łamach **INFORMATYKI**.

WACŁAW ISZKOWSKI

P.S. Zbitki prezentowanych tu poglądów i spostrzeżeń nie są oficjalnym stanowiskiem Redakcji ani też innej organizacji. Ich prezentacja ma służyć głównie pobudzeniu dyskusji na wybrany temat, nawet przez świadome wzburzenie moich przyszłych adwersarzy.

Konferencje

INFORMATYKA W SZKOLE VI

W dniach 17–20 września 1990 r. w Błężewku k. Kórnik odbędzie się kolejna, **VI Konferencja pn. „Informatyka w szkole”**. Tegoroczna konferencja będzie poświęconą przede wszystkim prezentacji oprogramowania wspomagającego nauczanie w szkołach podstawowych i średnich. Uczestników upatruje się w nauczycielach szkół podstawowych, średnich i wyższych, wizytatorach i metodach oraz pracownikach naukowo-badawczych. Do udziału w konferencji zostali zaproszeni m.in. wykonawcy systemów oprogramowania w programach RPI.09, RPI.14 i RRI.16. Organizatorzy zapraszają także firmy zajmujące się wytwarzaniem i dystrybucją sprzętu i oprogramowania dydaktycznego.

Organizatorami konferencji są: Instytut Informatyki Uniwersytetu Wrocławskiego, Ministerstwo Edukacji Narodowej i Ogólnopolska Fundacja Edukacji Komputerowej – Oddział Wlkp.

Zgłoszenia udziału w konferencji i oprogramowania do prezentacji należy przesyłać pod adresem: **Instytut Informatyki (Informatyka w Szkole VI), Uniwersytet Wrocławski, ul. Przemysłowa 20, 51-151 Wrocław.**

PRZEDSIĘBIORSTWO ZASTOSOWAŃ INFORMATYKI

meditronik

oferuje:

- systemy mikrokomputerowe
- programy aplikacyjne dla różnych dziedzin gospodarki (na życzenie wysyłamy katalog)
- poszukiwane komponenty elektroniczne
- interfejs do kamery video (opc. CCD) z bogatą biblioteką oprogramowania
- emulator Z80
- tester układów scalonych i pamięci
- programator BPROM
- asynchroniczny procesor komunikacyjny
- konwerter RS-232 – Centronics

instaluje:

- połączenia międzykomputerowe (XT/AT – ODRA/RIAD/IBM)
- systemy sieciowe (NOVELL)
- systemy wielodostępne (SCO Xenix 286, 386, Unix System V)

Jezeli jesteś autorem oryginalnego programu aplikacyjnego

- skontaktuj się z nami, będziemy pośredniczyć w sprzedaży Twojego programu, dbając o ochronę Twoich praw autorskich!

Nasz adres:

00-194 Warszawa, ul. Dzika 4
tel. (02) 635-22-63, 635-22-64,
fax (02) 635-21-95
tlx 816075 medi pl

Ogłoszenie w INFORMATYCE bezbłędną szansą trafienia do właściwego odbiorcy!

Miesięcznik INFORMATYKA jest od 25 lat **jedynym** w Polsce czasopismem naukowo-technicznym poświęconym wszystkim aspektom technologii informatycznej (sprzęt komputerowy, oprogramowanie, zastosowania). Taka pozycja czasopisma sprawia, że INFORMATYKA dociera nie tylko do najszerszych kręgów środowiska profesjonalnego, ale również do wszystkich zainteresowanych aktualnymi możliwościami tej technologii.

Ogłoszenie w INFORMATYCE to najskuteczniejszy sposób pozyskania nowych odbiorców i rozszerzenia rynku zbytu. Gwarantujemy Państwu największe efekty przy najniższych nakładach finansowych, ponieważ **ceny ogłoszeń w INFORMATYCE są znacznie niższe** niż w innych czasopismach:

• cała strona	500 000 zł	Dopłaty:	
• 3/4 strony	450 000 zł	• za dodatkowy kolor	+ 30%
• 2/3 strony	400 000 zł	• za I i IV stronę okładki	+ 100%
• 1/2 strony	300 000 zł	• za II i III stronę okładki	+ 50%
• 1/3 strony	200 000 zł	Rabaty:	
• 1/4 strony	150 000 zł	• za powtórzenie 3-6 razy	- 10%
• 1/8 strony	100 000 zł	• za powtórzenie ponad 6 razy	- 20%
• poniżej 1/8 strony	2 000 zł/cm ²	• za teksty o charakterze informacji techniczno-handlowej (artykuły)	- 50%

Zapewniamy szybką i terminową realizację ogłoszeń!

Nasz adres: INFORMATYKA, 01-552 Warszawa, Pl. Inwalidów 10, p. 128, tel. 39 14 34