Znajomość systemu operacyjnego przestała już być domeną bardzo wąskiej grupy specjalistów – analityków systemu lub zespołu realizacyjnego. Obecnie napisanie nowego systemu operacyjnego wydaje się dla wielu bardzo proste. A jednak, nowe systemy pojawiają się dość rzadko, częściej – starsze są przez lata udoskonalane i rozbudowywane, lecz zachowują błędy i ograniczenia dzieciństwa.

Zrealizowanie nowego systemu operacyjnego, a dokładniej funkcjonalnie pełnego oprogramowania systemowego dla danego typu komputera, jest jednak przedsięwzięciem wymagającym znacznych nakładów finansowych oraz długiego czasu wykonania. W historii informatyki w Polsce jest kilka przykładów stworzenia całkowicie od podstaw nowego systemu operacyjnego, np. SOM dla minikomputera MERA 400 oraz kilku systemów dla potrzeb akademickich w jednostkach instalacyjnych. Komputery seryjnie produkowane w Polsce były wyposażane w oryginalne systemy opracowane za granicą: ODRA 1300 – w Executive z George'm z ICL 1900, JS-RIAD w system OS i jego mutacje z IBM 360/370, a także SM-3 i SM-4 w system RSX-11 z PDP-11.

Obecnie wraz z rozwojem rynku mikrokomputerowego – opartego głównie na mutacjach IBM PC/XT i IBM PC/AT – rozpowszechnił się system operacyjny MS-DOS (PC-DOS), którego poznanie nie jest trudne. Jednak wraz ze wzrostem mocy obliczeniowej tych mikrokomputerów, spowodowanej zastosowaniem koprocesorów, pojemniejszych pamięci operacyjnych oraz dyskowych, a także – w związku z naciskiem użytkowników na tworzenie systemów wielodostępnych i sieci lokalnych ujawniły się wady tego systemu. Proste rozszerzanie funkcji MS-DOS przez nakładki systemów (jak np. Multilink), nie przyniosło oczekiwanych rezultatów. Pozostaje więc poszukiwanie innych rozwiązań. Jedynie system klasy Unix, coraz popularniejszy na rynku światowym, spełnia większość oczekiwań. Dla mikrokomputerów typu IMB PC możliwe jest, na przykład, wykorzystanie systemów Xenix lub QNX – oferowanych już na rynku polskim.

W aktualnej sytuacji polskiego rynku oprogramowania w relacji do rynku światowego istnieją jednak podstawowe trudności w użytkowaniu tych systemów. Ograniczenia przepływu technologii z Zachodu uniemożliwiają uzyskanie licencji i całkowicie blokują dostęp do kodu źródłowego, powodując tym samym niemożliwość jego lokalnych modyfikacji i rozszerzeń. Trzeba też powiedzieć, że dalsze bazowanie tylko na oprogramowaniu zachodnim, często kopiowanym nielegalnie, prowadziłoby nieuchronnie do dalszego opóźnienia w rozwoju informatyki w Polsce.

Stąd też szczególnej uwadze polecamy Czytelnikom prezentację systemu IPIX – zrealizowanego od podstaw w kraju. Wydaje się, że jest to znaczący fakt w rozwoju naszej informatyki, jeśli oprócz przejmowania produktów zachodnich mamy dostęp do produktów własnych. Popierając te działania, udział w pracach rozwojowych, a także praktyczne zastosowania, możemy mieć nadzieję, że łatwiej będzie pokonywać kolejny dystans w pogoni za postępem. Jednocześnie informujemy Czytelników, że Polskie Towarzystwo Informatyczne wspiera rozwój tego systemu, początkowo przez niezależną ocenę jego jakości, a w przyszłości może również finansowo w realizacji nowych modułów z jego środowiska. (WI)

DANUTA KRUSZEWSKA
Instytut Podstaw Informatyki
Polska Akademia Nauk
Warszawa

System operacyjny IPIX

IPIX jest nazwą systemu operacyjnego dla komputerów typu IBM PC, opracowanego w całości przez Zespół Technologii Oprogramowania Instytutu Podstaw Informatyki Polskiej Akademii Nauk. IPIX jest zgodny ze standardem Unix System V, zalecanym przez międzynarodową grupę X/OPEN (X/OPEN Portability Guide, Issue 2). Zgodność systemu IPIX z systemem Unix polega na ich funkcjonalnej zgodności wywołań i opcji poleceń systemowych oraz na pełnej zgodności wywołań funkcji systemowych i standardowych, zdefiniowanych dla języka C. Użytkownicy systemu Unix, Xenix lub podobnego, mogą więc bez żadnego przygotowania pracować pod nadzorem systemu IPIX. Programy napisane w języku C i działające pod kontrolą tych systemów można przenosić nawet między różnymi typami komputerów – oczywiście tylko w wersjach źródłowych.

IPIX – tak jak Unix – jest systemem uniwersalnym. Jest więc bazą, na której można zbudować dowolne oprogramowanie użytkowe, np. obsługę banków danych, obliczenia numeryczne czy przetwarzanie tekstów. Każdy system zgodny z Unixem ma wiele cech charakterystycznych dla tej klasy systemów operacyjnych.

Wielodostęp

System IPIX jednocześnie obsługuje wiele stanowisk pracy – konsolę (komputer główny) i terminale. Z każdego stanowiska użytkownik ma dostęp do wszystkich zasobów systemu i pracuje tak, jakby komputer i system były wyłącznie do jego dyspozycji.

Terminalem może być komputer typu IBM PC (z dowolnym emulatorem terminala, np. **xansi**, opracowanym specjalnie dla **IPIX-a**) ale może nią również być terminal alfanumeryczny (np. VT100). Terminal jest połączony z komputerem głównym przez łącze szeregowe RS232. W

niedalekiej przyszłości IPIX będzie także pracować w sieci.

Wielozadaniowość

System IPIX umożliwia równoczesne działanie wielu procesorów (zadań) i pozwala wykonywać jednocześnie wiele poleceń systemowych lub użytkowych (każde z nich również może być podzielone na procesy). Wielozadaniowość pozwala na bardzo efektywne wykorzystanie sprzętu, a tym samym zwiększa intensywność jego eksploatacji.

Mechanizm procesów wykorzystano przy tzw. przetwarzaniu w tle – IPIX umożliwia równoczesną pracę wsadową i konwersacyjną. Użytkownik może, na przykład, zainicjować w tle drukowanie wyników i uruchomić kompilator języka C dla kilku modułów programu, a jednocześnie redagować dokument (zanim poprzednie zadania się zakończą).

Mechanizm potokowy

System IPIX oferuje tzw. mechanizm potoków. Potok jest specjalnym rodzajem pliku, umiejscowionym na ogół w pamięci operacyjnej komputera. Jego cechą charakterystyczną jest to, że dane napływające są zapisywane zawsze na końcu pliku, a dane przeczytane – zawsze od początku – są usuwane z pliku.

Potoki najczęściej są wykorzystywane przy tworzeniu nowych poleceń z już istniejących. Taki potok poleceń jest ciągiem, w którym każde poprzednie polecenie przekazuje swoje wyniki bezpośrednio jako dane do następnego polecenia.

Hierarchiczny system plików

System plików IPIX-a jest drzewem o jednym korzeniu. Hierarchiczna struktura skorowidzów umożliwia tematyczne grupowanie plików. Jedną z właściwości tego systemu jest możliwość tworzenia dowiązań do zbiorów danych: jeden zbiór danych (program, tekst itp.) może być reprezentowany przez wiele plików o różnych lub tych samych nazwach w różnych skorowidzach.

Przez system plików IPIX umożliwia użytkownikom jednolitą obsługę wszystkich zasobów systemu: zbiorów danych, urządzeń wejścia i wyjścia, a nawet generatora dźwięków i pamięci operacyjnej.

Ochrony zasobów systemu i użytkowników

Ochronę zasobów systemu IPIX zrealizowano na podstawie mechanizmu prawa dostępu do plików. Superużytkownik (zarządca systemu) ma wszelkie uprawnienia dostępu do wszystkich zasobów systemu. Każdy inny użytkownik musi przedstawić się systemowi (przez podanie hasła) i od tej pory staje się właścicielem swoich plików oraz ma dostęp do tych plików systemowych, które udostępnił mu sam system lub jego zarządca. IPIX obsługuje również grupy użytkowników i tzw. resztę (potencjalni nowi użytkownicy).

Prawa dostępu określają możliwość odczytywania, zapisywania lub wykonywania pliku przez jego właściciela, grupę i resztę użytkowników. W ten sposób, przez niepowołanym dostępem można ochronić plik, skorowidz lub całe poddrzewo systemu plików.

Mgr DANUTA KRUSZEWSKA ukończyła w 1976 roku studia na Wydziale Matematyki i Mechaniki Uniwersytetu Warszawskiego (kierunek Informatyka). Pracuje w Instytucie Podstaw Informatyki PAN w zespole zajmującym się technologią tworzenia oprogramowania, który opracował i rozwija system operacyjny IPIX. Jest współautorką tłumaczenia książki pt. „Język C" (B. Kerninghan, D. Ritchie, „The C Programming Language") wydanej przez WNT w 1987 r.

Dlaczego powstał IPIX

To pytanie, na różnych imprezach i przez różnych ludzi, jest zadawane najczęściej. Przyzwyczailiśmy się już do argumentów w rodzaju:

„Chcecie konkurować z MS DOS-em? Nigdy nie zrobicie tak bogatego oprogramowania aplikacyjnego, jakie jest w Polsce dostępne!"

„Przecież jest w Polsce dostępny Xenix – system tej samej klasy – i na dodatek jest również dostępne zachodnie oprogramowanie aplikacyjne, lepiej zrobilibyście jakąś bazę danych!".

A my zwykle odpowiadamy tak.

Po pierwsze nie chcemy konkurować z MS-DOS-em!

●System ten ma rzeczywiście bogate oprogramowanie, ale jest systemem przeznaczonym dla komputerów osobistych (dla inżyniera, ale nie dla całej fabryki; dla sekretarki, ale nie dla całego biura; dla grafika, ale nie dla całego wydawnictwa); nie nadaje się do obsługi naprawdę dużych baz danych w naprawdę dużym przedsiębiorstwie (nawet gdy komputery są połączone w sieć).

●MS-DOS jest nieefektywny, bo „marnuje" możliwości sprzętu, tzn.:
– działa jednozadaniowo, jeśli program czeka na dane z dysku, to w tym czasie procesor odpoczywa, a mógłby wykonywać inne obliczenia (Intel 8086 – ok. miliona operacji na sekundę);
– „widzi" co najwyżej 640 KB pamięci operacyjnej, dlatego stosowanie go na PC/AT z pamięcią 2 MB jest marnotrawstwem pieniędzy (wbrew pozorom pamięć jest droga);
– bez sztuczek „widzi" tylko 32 MB dysku Winchester;
– wszystkie najbardziej znane pakiety, np. graficzne, działają poza systemem (bezpośrednio sięgają do możliwości sprzętu).

●Kolejne wersje MS-DOS-a coraz bardziej przypominają system Unix. Na dodatek, system ten jest w wielu wypadkach użytkowany prawem kaduka – ogromna większość jego użytkowników nie kupiła go oficjalnie.

Po drugie, gdy zaczynaliśmy prace nad systemem IPIX, system Xenix nie był jeszcze tak rozpowszechniony w Polsce, jak dziś.

Oprócz mechanizmu uprawnień system zapewnia możliwość blokowania dostępu do danych pliku (nawet do jednego bajtu), to znaczy pozwala określić: co, przez kogo i w jaki sposób jest blokowane oraz sprawdzić, czy określony obszar w pliku jest zablokowany i jaki jest rodzaj tej blokady.

Konwersacyjny interpretator poleceń SHELL

Komunikacja użytkownika z systemem odbywa się za pośrednictwem interpretatora poleceń **Shell** – jego zadaniem jest wykonywanie poleceń użytkownika. Do niego także należy (na życzenie użytkownika) definiowanie lub zmiana kierunku przesyłania danych, łączenie poleceń w grupy lub potoki, wykonywanie poleceń w tle itp.

Na ogół Shell wykonuje polecenia w trybie konwersacyjnym, ale umożliwia też użycie proceduralnego języka programowania rozbudowanych plików poleceń wsadowych (skrypty Shella), wykorzystujących inne, już istniejące polecenia. Korzystanie z języka Shella eliminuje w wielu wypadkach konieczność pisania nowych programów.

Zestaw poleceń systemowych i użytkowych

System IPIX zapewnia szereg podstawowych usług, których wywołania są na-

Prace nad systemem IPIX są prowadzone w Instytucie Podstaw Informatyki PAN od 1984 roku. Wersja przeznaczona dla komputerów typu IBM PC/XT powstała dzięki zamówieniu Krakowskiej Fabryki Aparatów Pomiarowych MERA-KFAP, która w przeważającej części sfinansowała (ważne!) prace nad systemem przeznaczonym dla produkowanego przez siebie komputera KRAK-86. Wynik tych prac – IPIX – jest rozprowadzany przez MERA-KFAP razem z ich komputerem! IPIX jest również oferowany przez IPI PAN jako samodzielny produkt rynkowy dla użytkowników i producentów innych komputerów zgodnych z IBM PC.

Na samym początku pracy założyliśmy, że musi ona być w całości wykonana w Polsce i że nie będzie polegała na „wypruwaniu metek" (zwrot świetnie ujmujący tendencje na polskim rynku). Pomijając jednak prawne i etyczne aspekty naszej decyzji (w Polsce nie ma skutecznej ochrony prawnej ani oprogramowania zachodniego, ani krajowego), chcemy zwrócić uwagę na jej aspekt czysto pragmatyczny. Dzięki tej pracy jesteśmy zespołem ekspertów panujących nad systemem pod każdym względem. Możemy zatem:

- z powodzeniem instalować IPIX-a na komputerach niekoniecznie dokładnie zgodnych z IBM-owskim pierwowzorem (instalacja systemu Xenix na nietypowym sprzęcie czasami jest w ogóle niemożliwa, a często wymaga uruchomienia własnych programów obsługi niektórych urządzeń zewnętrznych);
- natychmiast reagować na uwagi użytkowników, dotyczące jakości systemu, tzn. usuwać błędy, zmieniać stosowane algorytmy itp. (nie ma przecież programu bez wad, Xenix też ma błędy);
- rozwijać, a przede wszystkim wspomagać rozwijanie systemu w dowolnie wybranych kierunkach zastosowań; IPIX nie musi być uniwersalny; IPIX (lub tylko jego jądro) może być częścią systemu aplikacyjnego zrobionego „pod klucz", a algorytmy IPIX-a możemy dostosować do szczególnych wymagań lub dla zwiększenia efektywności i aplikacji.

Mieszkamy i pracujemy w Polsce – można do nas zatelefonować lub przyjechać, wyjaśnimy wątpliwości, pomożemy (np. w postaci konsultacji czy kursu) w początkowym okresie eksploatacji systemu.

Ostatnia, ale chyba najważniejsza zaleta – możemy przenosić system na inne komputery, niekoniecznie oparte na procesorze Intel. Co więcej – oprogramowanie dla IPIX-a, które do tego czasu powstanie, będzie można bez istotnych problemów przenieść na ten nowy komputer (oczywiście w wersji źródłowej). Ta możliwość stanie się niezwykle istotna w chwili, gdy polski przemysł elektroniczny naprawdę będzie produkował polskie komputery i będzie chciał nimi handlować poza granicami kraju.

Uczestniczymy z IPIX-em w różnych wystawach, demonstrujemy jego możliwości na różnych konferencjach, prowadzimy rozmowy handlowe i dotyczące współpracy z różnymi instytucjami – zarówno państwowymi, jak i prywatnymi, w kraju i za granicą. Udało się nam sprzedać kilka egzemplarzy systemu. Głęboko wierzymy, że nowy komputer Elwro 801 AT najlepiej będzie wykorzystany przez system IPIX. A jednak nadal słyszymy to pytanie: **Dlaczego powstał IPIX?**

W pracach nad systemem IPIX uczestniczyli lub uczestniczą następujący członkowie Zespołu Technologii Oprogramowania: doc. dr Jan Borowiec – Szef Zespołu, mgr inż. Elżbieta Jezierska-Ziemkiewicz, mgr Danuta Kruszewska, mgr Marek Kruszewski, mgr Stanisław Kruszewski, mgr Dariusz Kupiecki, dr Andrzej Łukasiewicz, mgr Władysław Majerski, mgr inż. Mirosław Malicki, mgr Tadeusz Paprzycki, mgr inż. Jacek Surma, dr Jan Walasek – Zastępca Szefa, mgr inż. Tadeusz Wilczek, mgr Julian Winiewski, mgr inż. Andrzej Ziemkiewicz.

Historia Unixa

Z systemami operacyjnymi klasy Unix (IBM/IX, Xenix, QNX) większość informatyków w Polsce zetknęła się w momencie, kiedy MS-DOS miał już ugruntowaną pozycję. Stąd częste przekonanie, że Unix jest produktem późniejszym od MS-DOS-u, co z kolei jest źródłem licznych nieporozumień, ponieważ chronologicznie było dokładnie na odwrót.

Wszysko zaczęło się w 1968 roku, kiedy to w Bell Laboratories (w USA) spotkali się Ken Thompson i Dennis Ritchie. Thompson przybył z Berkeley, gdzie rozwijano wówczas system operacyjny SDS930, zaś Ritchie – z Harvardu, gdzie zajmował się matematyką stosowaną. Grupa badawcza, w której skład weszli, zwróciła się do kierownictwa o środki na odpowiedni komputer do prowadzonych prac, jednakże takich środków nie otrzymała. Po dyskusjach Ritchie i Thompson postanowili wykorzystać dostępny minikomputer PDP-7 i stworzyć dla niego odpowiedni system operacyjny. Był to minikomputer o architekturze zbliżonej do współczesnych mikrokomputerów, ale jego oprogramowanie składało się jedynie z asemblera i programu ładującego. W 1969 roku system operacyjny we wstępnej wersji był gotowy. Miał hierarchiczny system plików, możliwość powoływania zadań współbieżnych (w terminologii UNIXa – procesów) i obsługiwał tylko jednego użytkownika.

Nazwę Unix wymyślił Kerninghan w 1970 roku. Miała to być nazwa ironiczna, utworzona przez analogię do nazwy wieloużytkowego systemu Multics, choć jeszcze w tym samym roku Unix potrafił obsługiwać dwóch użytkowników. Początki Unixa były więc bardzo skromne, ale nawet w tej pierwotnej wersji cechowała go niezwykła wprost elegancja rozwiązań – owo specyficzne połączenie prostoty i wyrafinowania, mocy i zwartości. Do dziś Unix jest oceniany w kategoriach tyleż użytkowych co, estetycznych.

System od razu zdobył uznanie innych pracowników Bell Laboratories. Został wykorzystany do różnych prac, przede wszystkim do przetwarzania tekstów.

Dalsza historia Unixa jest następująca:

1971 – przeniesienie na komputer nieco większej mocy, PDP 11/20,
1972 – włączenie mechanizmu potoków,
1973 – Thompson skończył opracowywanie języka C i system został przepisany na ten język,
1975 – pierwsza wersja licencjonowana;

W jakiś czas potem, wykorzystując programy źródłowe w języku C, przeniesiono system na komputer Interdata 8/32, który ma architekturę tak odmienną od serii PDP, iż nie było wątpliwości, że Unix jest systemem naprawdę przenośnym. Potem dokonano licznych innych przeniesień na komputery różnych typów i generacji. Ich podstawą były programy źródłowe w języku C udostępnione przez AT&T. Poza tym powstały liczne systemy klasy Unix lub podobne, realizowane od podstaw. Obecnie istnieje ponad 100 takich systemów. Sama nazwa Unix jest zastrzeżona dla systemów dostarczanych przez AT&T.

1979 – pierwsza licencjonowana wersja handlowa,

1983 – Unix System V, opublikowana norma systemu, ustalona przez AT&T,

1984 – Ritchie i Thompson dostają za Unix doroczną nagrodę Turinga za rok 1983 (przyznawaną przez Association for Computing Machinery).

Od 1981 roku na historię UNIX-a nakłada się historia mikrokomputera IBM PC. Firma IBM zwróciła się do firmy Microsoft o skonstruowanie odpowiedniego systemu operacyjnego dla tego mikrokomputera. Po rozmowach szef firmy Microsoft – Bill Gates – przyjął zamówienie i powstała pierwsza wersja systemu MS-DOS (przez IBM nazywanego PC-DOS). Jednocześnie jednak Gates podjął decyzję, że przyszłym systemem dla IBM PC będzie system klasy Unix. Zwrócił się więc do AT&T i kupił wersję źródłową Unixa; trochę później powstał system Xenix. Nie jest jasne, dlaczego Gates wówczas nie przyjął, że MS-DOS będzie podzbiorem przyszłego Xenixa – choćby bardzo ograniczonym. Tymczasem rozwinęły się komputery IBM PC i odpowiednio firma Microsoft rozwinęła MS-DOS, zresztą przez kolejne włączanie różnych mechanizmów Xenixa. MS-DOS stał się sukcesem rynkowym, ale też hamował popyt na Xenixa. Sam MS-DOS nie może przez kolejne rozszerzenia przekształcić się w Xenix, gdyż nie pozwalają na to wcześniej przyjęte ustalenia.

Przełom nastąpił wówczas, gdy pojawił się PC/AT – mikrokomputer, który może być w pełni wykorzystany przez Xenix lub system tej klasy. Pojawiły się jednak trudności z innej strony. Microsoft włączył do swojego Xenixa kilka istotnych rozszerzeń (blokowanie rekordów, pamięć współdzielona, semafory itd.). Firma AT&T włączyła te rozszerzenia do swojej normy Unix System V, ale zaimplementowała je w odmienny sposób. Biorąc pod uwagę potęgę AT&T, nie było to wygodne dla Microsoftu. Ostatecznie Gates zdecydował, że należy doprowadzić do zgodności Xenixa z Unixem (System V) i taką wersję opracowała firma współpracująca z Microsoftem – Santa Cruz Operation.

W 1986 roku powstała organizacja X/OPEN, grupująca znaczące przedsiębiorstwa komputerowe, której celem jest zalecanie istniejących norm w dziedzinie oprogramowania. Jako normę systemu operacyjnego zaleca się Unix System V, a dla języka C – normę ANSI. Inne zalecenia odnoszą się do baz danych, operacji ekranowych itd. Unix jest pierwszym systemem objętym normalizacją w tak szerokim zakresie. **(DK)**

zwane poleceniami systemowymi. Należą do nich m. in. polecenia obsługujące system plików, wspomagające powstawanie nowych poleceń, zarządzające wykonywaniem uruchomionych zadań itp.

Część usług systemu jest realizowana przez programy użytkowe odgrywające rolę typowych programów usługowych, np. ekranowy edytor tekstów (tworzenie plików tekstowych), obsługa plików tekstowych (porównywanie, sortowanie itp.) czy obsługa plików w formacie systemu MS-DOS. Przez polecenie tego rodzaju rozumie się program (lub skrypt Shella) napisany przez użytkownika. Z punktu widzenia systemu IPIX nie ma żadnej różnicy w traktowaniu poleceń systemowych i programów użytkowych.

W systemie IPIX takich poleceń jest ok. 100 (standard zawiera ok. 120 opisów poleceń, przy czym niektóre polecenia nie są obowiązkowe).

Kompilator języka C

Głównym językiem programowania w systemie IPIX jest język C. Jego kompilator akceptuje pełny język C według raportu: B. W. Kerninghan, D. M. Ritchie, The C Programming Language, Prentice-Hall, 1982 (polskie tłumaczenie: Język C, WNT, 1987). Kompilator tworzy program według jednego z czterech modeli pamięci. Standardowa biblioteka funkcji wejścia i wyjścia (dla każdego modelu pamięci) jest całkowicie zgodna ze standardem Unix System V.

* * *

Jednocześnie w kilku ośrodkach prowadzi się intensywne prace nad rozwojem systemu IPIX. Celem tych prac jest rozszerzenie i wzbogacenie możliwości systemu. Wród nich na szczególną uwagę zasługują:

- nowa dedykowana wersja IPIX dla komputerów opartych na procesorze Intel 80386,
- kompilator języka Pascal,
- oryginalny asembler o składni języka wysokiego poziomu,
- pakiety funkcji obsługujących ekran i urządzenia graficzne,
- pakiety funkcji wspomagających zarządzanie bazą danych (ISAM),
- język SQL dla tworzenia relacyjnych baz danych.

Ponadto, z planowanych prac warto wymienić opracowania kompilatorów innych nowoczesnych języków programowania, jak Loglan i Prolog oraz narzędzi do pracy w sieci.

Rysunki SZYMON KOBYLIŃSKI

W skrócie • W skrócie • W skrócie • W skrócie

□ Zespoły badawcze firmy IBM i AT&T mają pracować w ośrodku badawczym IBM znajdującym się w La Gaude we Francji. Ich celem jest określenie zapotrzebowania rynku europejskiego na inteligentne usługi sieciowe, obejmujące również sieci prywatne.

*

□ Oddział systemów testujących (Sentry Test Systems Division) przedsiębiorstwa Schumberger Ltd. oferuje nowy tester kostek pamięci dynamicznej CMOS. Tester ten, o nazwie S90, może badać jednocześnie 32 kostki z częstotliwością 50 MHz (dotychczasowe testery mogły kontrolować maksymalnie 16 kostek równolegle). Jest on przystosowany do układów o submikronowej strukturze, takich jak kostki pamięci statycznej o pojemności 256 K bitów i dynamicznej o pojemności 1 M bitu. Ma cztery głowice pomiarowe, z których każda może sondować 8 mikroukładów dynamicznych lub cztery statyczne. W zależności od konfiguracji cena systemu waha się od 250 do 600 tysięcy dolarów, co oznacza 19 375 dolarów za jedno stanowisko testujące, a więc taniej niż w dotychczasowych testerach.

*

□ Firma Texas Instruments Inc. ogłosiła parametry swego procesora sygnałów cyfrowych – TM9320C10. Wykonuje on 6,4 miliona rozkazów na sekundę, a jego część związana z układami CMOS działa z maksymalną częstotliwością zegara 25,6 MHz i jest w pełni zgodna ze standardowym procesorem 32010 realizowanym na układach z kanałem n, który jest o 25% wolniejszy. Mimo dużej szybkości kostka CMOS ma typowy pobór prądu tylko 35 mA, a więc pobór mocy wynosi tylko piątą część tego, co w układzie 32010. Kostka 32010 ma czterdziestokońcówkową, plastikową obudowę z dwurzędowymi wyprowadzeniami (DIL). W partiach po 100 sztuk jest sprzedawana po 60 dolarów. Jednocześnie firma Texas rozszerza swą rodzinę liniowych układów CMOS o dwa nowe układy regulatorów czasowych i dwa nowe komparatory różnicowe. Oba działają z pojedynczego zasilacza o napięciu zaledwie 1V.

W skrócie • W skrócie • W skrócie • W skrócie