

racji, która pojawiła się w latach 1960/1961 z chwilą rozpoczęcia sprzedaży maszyn typu IBM 1401 i RCA 301. Natomiast w latach 1964/1965 wprowadzono nową technikę, realizowaną za pomocą tzw. układów scalonych (*integrated circuit*), które bardzo zminiaturyzowały gabaryty układów elektronicznych maszyn. Maszyny budowane w tej technice zaliczono do III generacji. Zaliczyć do nich można maszyny typu IBM 360, IBM 1130, RCA — Spectra 70, Univac 1108, System 4, Gamma 140, ICL 1900A i inne. Warto podkreślić, że w wielu wypadkach zakwalifikowanie maszyn do odpowiednich generacji jest trudne, czasem nawet mówi się o maszynach 2,5 generacji, np. o maszynie typu GE 400, realizowanej w technice *hi-pack*. Mówi się również o maszynach 3,5 generacji, np. NCR Century Series 600, której stosunek ceny do skuteczności przetwarzania jest tak korzystny w porównaniu z innymi maszynami III generacji, że zalicza się je do nieco wyższej generacji.

W wielu wypadkach kryterium podziału generacji według techniki zastosowanej w budowie jest nie dość precyzyjne. Bowiem z punktu widzenia układu logicznego maszyny, niektóre maszyny III generacji nie różnią się poza techniką od maszyn II generacji; szczególnie ma to miejsce przy porównaniu maszyn zapowiadanej IV i III generacji. Wydaje się, że oprócz rozróżniania techniki warto zwrócić uwagę na różnice występujące w układzie logicznym maszyny. Z tego ostatniego względu występuje dość duża różnica między maszynami II i III generacji w organizacji hierarchicznej pamięci poprzednio omówionej.

Na ryc. 2.18 przedstawiamy pakiety trzech pierwszych generacji komputerów matematycznych, które poza różnicą w niezawodności i prędkości działania znacznie różnią się gabarytami. Na ryc. 2.19 przedstawiamy schemat procesu technologicznego powstawania obwodu scalonego-hybrydowego produkcji IBM dla maszyn 360 i 1130, natomiast na ryc. 2.20 — scalony obwód monolityczny produkowany przez ICL dla maszyn System 4.

2.3. Organizacja przetwarzania w systemach wielodostępnych (abonenckich)

Rozwój zestawów komputerowych w latach siedemdziesiątych, prowadzi w kierunku budowy: a) minikomputerów jako bardziej specjalizowanych i tańszych oraz b) komputerów wielkich, a nawet i superkomputerów (cena ponad 10 mln dol.). W tym punkcie zwracamy uwagę na sposoby wykorzystania komputerów większych (należących do dużych, wielkich i super⁸) przez liczną grupę użytkowników liczących wspólnie na tym samym komputerze. Wykorzystanie tego typu komputerów znajduje (por. ryc. 2.21) miejsce w:

⁸ Por. A. Targowski, *Organizacja ośrodków obliczeniowych*, Warszawa 1971.

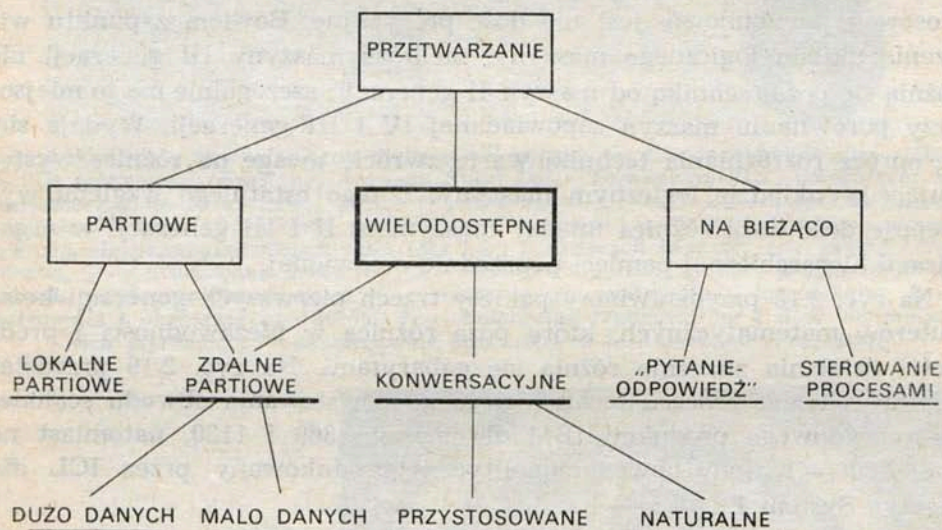
a) przetwarzaniu partiowym, kiedy dane mogą być wprowadzane w miejscu zainstalowania komputera (*local batch processing*) albo w miejscu oddalonym od niego (*remote batch processing* albo *remote job entry* (RJE) przy wykorzystaniu końcówki komputerowej i sieci transmisji danych,

b) przetwarzaniu wielodostępnym (*time sharing*):

— kiedy dane wprowadzane są partiami z odległych miejsc (zdalne przetwarzanie partiowe),

— konwersacyjnym albo interakcyjnym, które polega na stałym kontakcie użytkownika z maszyną w całym cyklu obliczeniowym,

c) przetwarzaniu na bieżąco w systemach pytanie-odpowieź albo w systemach sterowania procesami technologicznymi (rzadko spotykane wykorzystanie tego typu komputerów).



Ryc. 2.21. Miejsce przetwarzania wielodostępnego w zakresie najpopularniejszych systemów wykorzystujących większe komputery

Zdalne przetwarzanie partiowe może dotyczyć dużego wolumenu danych (*hi-speed remote job entry* albo HSRJE) albo małego (*lospeed* albo LSRJE). Przetwarzanie konwersacyjne może odbywać się przy użyciu języka programowania naturalnego typu np. APL lub przystosowanego do potrzeb danego typu obliczeń np. BASIC, FORTRAN. Z tego przeglądu zastosowań większych komputerów wynika, że przetwarzanie wielodostępne wykorzystywane jest do zdalnego przetwarzania partiowego i konwersacyjnego. Dla tego sposobu przetwarzania charakterystyczne jest, że użytkownik nie zajmuje się problematyką eksploatacji zestawu komputerowego. Jego zainteresowania skupiają się wokół problemu, który rozwiązuje za pomocą środka, jakim jest końcówka. W latach

siedemdziesiątych przewiduje się wykorzystanie końcówek do następujących przykładowo wybranych zastosowań:

— kadra kierownicza wykorzystywać będzie końcówki ekranowe w systemie pytanie-odpowieź,

— sekretariaty pisać będą korespondencję na końcówkach (maszyny do pisania), kopie będą datowane przez komputer i w jego pamięci archiwizowane, oryginały będą wystawiane automatycznie na końcówkach odbiorców, lub gdy ci ich nie posiadają, na końcówkach lokalnego urzędu pocztowego,

— bibliotekarze wysyłać będą informacje do użytkowników bezpośrednio na ich końcówki (i vice versa),

— konserwatorzy np. komputerów będą wysyłali zapytanie przez końcówki do centrali, a w wypadku awarii sprzętu w celu otrzymania dokładnych porad wraz z interpretacją graficzną postępowania w wypadku usuwania awarii itp.

Stosunkowo duże osiągnięcia w zakresie przetwarzania konserwacyjnego uzyskała firma CONTROL DATA, stosująca maszyny CDC 3000 i CDC 6000 w sieci obliczeniowej CYBERNET. Podobną sieć zorganizowała firma GENERAL ELECTRIC wykorzystując komputery GE 200 w sieci pod nazwą MARK I i komputery GE 600 w sieci pod nazwą MARK II. Firma IBM uruchomiła wiele tego typu sieci poczynając od QUICTRAN (komputer IBM 7040) przez RAX, CALL, CP/CMS, CPS, CRJE, ITF, TSO aż do TSS wykorzystując komputery serii 360 i 370.

2.3.1. Użytkownicy końcówek komputerowych

Projektując system przetwarzania wielodostępnego należy zbadać typowe profile potencjalnych użytkowników określając ich zapotrzebowanie na określone (typowe) zadania obliczeniowe. Biorąc pod uwagę zainteresowania techniką obliczeniową można wyróżnić pięć grup użytkowników końcówek komputerowych.

1. Użytkownik — „nie twórca”; o zasadach działania komputera wie tyle samo co o telefonie choć oba media stosuje. Wykorzystując końcówkę nie tworzy żadnych problemowych sytuacji z wyjątkiem wprowadzania pytania, mieszczącego się w dokładnie zaprojektowanej procedurze i otrzymuje odpowiedzi w bardzo czytelnej formie. Przykładem tego typu użytkownika może być agent linii lotniczych wystawiający bilet w ramach komputerowego systemu rezerwacyjnego.

2. Student, stosujący końcówkę jako narzędzie pomagające w opracowywaniu: ćwiczeń, sprawozdań laboratoryjnych, projektów itp. Potrzeby tego typu użytkownika są trudne do zdefiniowania, najbardziej przeciętne polegają na stosowaniu: różnych języków programowania, instruowania przez komputer w sprawach korzystania z systemu rozwiniętych funkcji diagnostycznych przetwarzanych programów.

3. Naukowiec; nie jest informatykiem, ale jego podejście do wykorzystania komputera jest bardziej przemyślane niż użytkowników poprzednich grup. Jego potrzeby sprowadzają się do: niewielkiej pojemności pamięci, podstawowego języka programowania oraz standardowych programów z zakresu badań operacyjnych. Użytkownicy należący do tej grupy najchętniej posługują się językiem programowania BASIC i przyczynili się walcie do sukcesów ośrodków usługowych.

4. Użytkownik pakietów programowych do określonych zastosowań; jest specjalistą-problemistą, który w odróżnieniu od naukowca nie zna nawet najprostszych metod programowania. Natomiast jego potrzeby obliczeniowe są dokładnie zdefiniowane i mogą być zaspokojone dostępem do pakietów programowych, np. typu systemu symulacyjnego GPSS czy systemu harmonogramowania PERT.

5. Programista wykorzystujący system wielodostępny do uruchamiania programów wykorzystywanych np. w przetwarzaniu partiovym.

Biorąc pod uwagę wyposażenie systemów wielodostępnych można rozróżnić trzy klasy użytkowników.

Do klasy A należą użytkownicy eksploatujący jedną maszynę w jednakowy sposób i za pomocą tego samego sprzętu zewnętrznego. Mogą



Ryc. 2.22. Kalkulator stołowy CP 6060 (Fot. CDC)

to być agenci systemu rezerwacji miejsc lotniczych, korzystający z jednakowych urządzeń lub problemisci korzystający np. z kalkulatorów CDC 6060 (por. ryc. 2.22). Do jednej maszyny może być podłączonych zdalnie, np. 128 takich urządzeń.

Do klasy B należą użytkownicy wykorzystujący jedną maszynę do rozwiązywania różnych problemów i stosujący różne urządzenia zewnętrzne, np. dalekopisy, kodopisy, ekranopisy i inne monitory (por. ryc. 2.23). Do tej klasy użytkowników zalicza się m.in. naukowców, używających system MAC w Massachusetts Institute of Technology.



Ryc. 2.23. Monitor wyjściowo-graficzny CD (Fot. CDC)

Do klasy C należą użytkownicy wykorzystujący system wielomaszynowy do rozwiązywania różnych problemów i stosujący różne urządzenia zewnętrzne; jako satelitarne mogą być stosowane np. maszyny serii CDC 3000 w połączeniu z maszynami serii CDC 6000.

2.3.2. System rozkazowy użytkownika

Sprawą zasadniczą w zaakceptowaniu przez użytkownika systemu wielodostępnego jest jego pierwsze wrażenie, jakie uzyskuje podczas pierwszego kontaktu z końcówką. Użytkownik końcówki nie jest zawodowym programistą i nie zaakceptuje dokładnych reguł występujących w tradycyjnym programowaniu. Jego pierwsze wrażenie będzie zależało od demonstracji wykonanej przez użytkownika doświadczonego w korzystaniu z końcówki albo od dobrze pod względem metodycznym opracowanego podręcznika. Lecz najlepszą gwarancją w każdym wypadku jest zapewnienie łatwego kontaktu użytkownik—końcówka, który zapewnia dobry język rozkazowy użytkownika (*command language*), którym posługuje się przy obsłudze końcówki.

Aby włączyć końcówkę do systemu abonenckiego, naciska się określony klawisz, który włącza sygnał linii telefonicznej. Następnie nakręca się numer telefoniczny, pod którym umieszczony jest dany system abonencki, założmy, że jest to MARK II (GE). Z chwilą uzyskania połączenia (numer może być zajęty, jeżeli w danej chwili zgłosiło się więcej użytkowników, niż system akceptuje, zwykle liczba końcówek przewyższa dopuszczalną liczbę współbieżnie liczących użytkowników) system MARK II uruchamia serię pytań (systemu) i odpowiedzi (użytkownika), które identyfikują użytkownika i jego zadanie obliczeniowe. Użytkownik podaje informacje o: a) sobie, tzn. symbol identyfikacyjny i hasło (*password*), b) języku programowania, w którym odbędzie się liczenie, c) zbiorze czy nowy, czy stary, d) nazwie zbioru.

Symbol użytkownika i hasło przydziela administrator systemu wielodostępnego. Symbol użytkownika jest nazwą, pod którą system będzie śledził sposób i czas eksploatacji komputera. Hasło, które może być często zmieniane ma zagwarantować, że tylko ten a nie inny użytkownik może prowadzić obliczenia posługując się danym symbolem identyfikacyjnym. Dzięki temu użytkownik ma pewność, że ktoś inny nie liczy na jego rachunek. Aby ktoś obcy nie odczytał hasła, które użytkownik wprowadza z pulpitu końcówki, stosuje się odpowiednio zabezpieczające procedury. Każdy znak hasła po wydrukowaniu jest momentalnie automatycznie przeбитý innym znakiem uniemożliwiając w ten sposób odczyt osobie postronnej. Może być również stosowana taka procedura, która polega na tym, że palcząc hasło system powstrzymuje karetkę od pisania znaków na papierze tabulogramowym.

Po otrzymaniu połączenia telefonicznego system drukuje na końcówce swoją identyfikację (symbol, czas, datę):

MARK II 008-706 17:13 20 AUG 71,

następnie system stawia pytania, a użytkownik wypisuje odpowiedzi (litery podkreślone):

Konwersacja między użytkownikiem a systemem abonenckim odbywa się w ramach systemów:

- rozkazów (*system commands*),
- zapytań (*system request*),
- odpowiedzi na pytania (*responses to system requests*),
- komunikatów,
- języków programowania.

System rozkazów i odpowiedzi na pytania jest inicjowany przez użytkownika. System zapytań i komunikatów jest inicjowany przez dany system wielodostępny.

System zapytań wskazuje, że dany system wielodostępny wymaga dodatkowych informacji, aby rozpocząć (lub dalej kontynuować) liczenie. System komunikatów wskazuje, że system wielodostępny ma niekom-

USER NUMBER — WE0560, ORNAK			
			symbol
			hasło
PROJECT ID — KRATOWNICA 32			
			nazwa zadania
			obliczeniowego
SYSTEM — BASIC			
			nazwa języka
			programowania
NEW OR OLD — OLD			
			nowy czy stary zbiór
ENTER FILE NAME — MOMENTY			
			nazwa zbioru
READY			system gotowy do
			liczenia
RUN			licz

pletne informacje albo wymaga odpowiednich procedur korygujących proces liczenia.

MARK II posiada 57 różnych rozkazów i 60 różnych rodzajów zapytań, które wykorzystuje się do sterowania przetwarzaniem. Użytkownik posiada podręczny spis rozkazów i komunikatów, tak że nie musi znać ich na pamięć. Na przykład, aby utworzyć kartotekę użytkownik pisze na końcu rozkaz CREATE (twórz), następnie system stawia pytania:

PASSWORD (hasło)?	JUNCZYK
SEQUENTIAL OR RANDOM?	SEQUENTIAL
(organizacja zbioru sekwencyjna lub wrywkowa)	
RECORD SIZE IN WORD?	1
(liczba słów w zapisie)	
NUMBER OF RECORDS?	945
(liczba zapisów)	

Jeżeli użytkownik błędnie napisał rozkaz, wtedy system może wysłać do niego komunikat WHAT? (co?). Zakończyć liczenie można przez użycie rozkazu, np. w systemie IBM TSS/360 — LOGOFF, po którym system podaje informację — „przyjąłem” oraz określa czas wykorzystania czasu procesora i urządzeń zewnętrznych.

2.3.3. Języki programowania

System rozkazów i komunikatów służy do sterowania przetwarzaniem w układzie użytkownik—komputer, natomiast język programowania służy do sterowania przetwarzaniem w układzie dane—zbiory—wyniki. Z punk-

tu widzenia sposobu tłumaczenia programów wyróżnia się translatory języków programowania typu:

— „load — and go” (po przetłumaczeniu natychmiast odbywa się wykonanie programu)

— interpretacyjnego (po przetłumaczeniu rozkazu języka użytkownika odbywa się wykonanie programu)

— występującego w przetwarzaniu partiovym.

Biorąc pod uwagę samą budowę języka programowania, w systemach wielodostępnych wykorzystywane są:

a) tradycyjne języki programowania jak np.: FORTRAN, ALGOL, PL/I, COBOL, RPG, AUTOCODE typu ASSEMBLER,

b) specjalne języki konwersacyjne: BASIC, BRUIN, APL,

c) problemowo zorientowane języki: SNOBOL, LISP.

Wprowadzanie programu może odbywać się albo czytnikiem kart lub taśmy dziurkowanej, albo bezpośrednio z pulpitu końcówki. Każdy rozkaz programu poprzedza jego numer kolejny np. 230 READ D (J,3), D (J,2), gdzie 230 jest właśnie tym numerem. Poleca się numerować rozkazy liczbami co dziesięć np. 10, 20, 30, 40 itd. po to, by w razie potrzeby można było włączyć do programu nowe rozkazy, np. 12, 13 itd. Po wprowadzeniu programu użytkownik poleca jego ponowne wydrukowanie na końcówce, używa wtedy w systemie MARK II rozkazu LIST, albo poleca przechować program, pisze wtedy SAVE lub żąda jego wykonania, wtedy pisze RUN.

2.3.4. Zbiory

Każdy użytkownik systemu wielodostępnego ma prawo do utworzenia i przechowywania w systemie własnego zbioru danych i własnych programów. Aby uchronić zbiory przed wykorzystywaniem ich przez niepożądanego użytkownika, system posiada system kontroli dostępu użytkownika do całego systemu oraz do poszczególnych zbiorów i programów. Kontrola dostępu użytkownika odbywa się przez symbol użytkownika i hasła (por. pkt 2.3.2). System kontroli dostępu do zbiorów musi uwzględniać fakt, że zbiory mogą być dostępne:

- 1) tylko dla jednego użytkownika,
- 2) dla określonej grupy użytkowników,
- 3) dla wszystkich użytkowników.

Następnie dostęp do zbiorów może polegać na zezwoleniu tylko:

- 1) na czytanie zbioru,
- 2) czytanie i zapisywanie danych w zbiorze,
- 3) czytanie, zapisywanie i wymazywanie danych w zbiorze.

Biorąc pod uwagę tę charakterystykę dostępności do zbiorów, system musi chronić zbiory przed dostępem: a) wszystkich użytkowni-

ków, b) wybranej grupy użytkowników lub zezwalać na dostęp, c) wszystkim.

Tworzenie zbiorów przez użytkownika polega na określeniu nazwy zbioru, którą system wprowadzi do KATALOGU NAZW ZBIORÓW wraz z symbolem identyfikacyjnym użytkownika. Następnie użytkownik używając rozkazu EDIT WIERSZ wprowadza dane zbioru w tzw. wierszach, które są ponumerowane np.:

000100 BYĆ ALBO NIE BYĆ OTO JEST PYTANIE

000200 KTÓRE NIE STRACIŁO NIC NA SWEJ WAŻNOŚCI

W wypadku uaktualniania zbioru, użytkownik może włączyć nowe dane w wierszu 250, wtedy używa rozkaz (włącz):

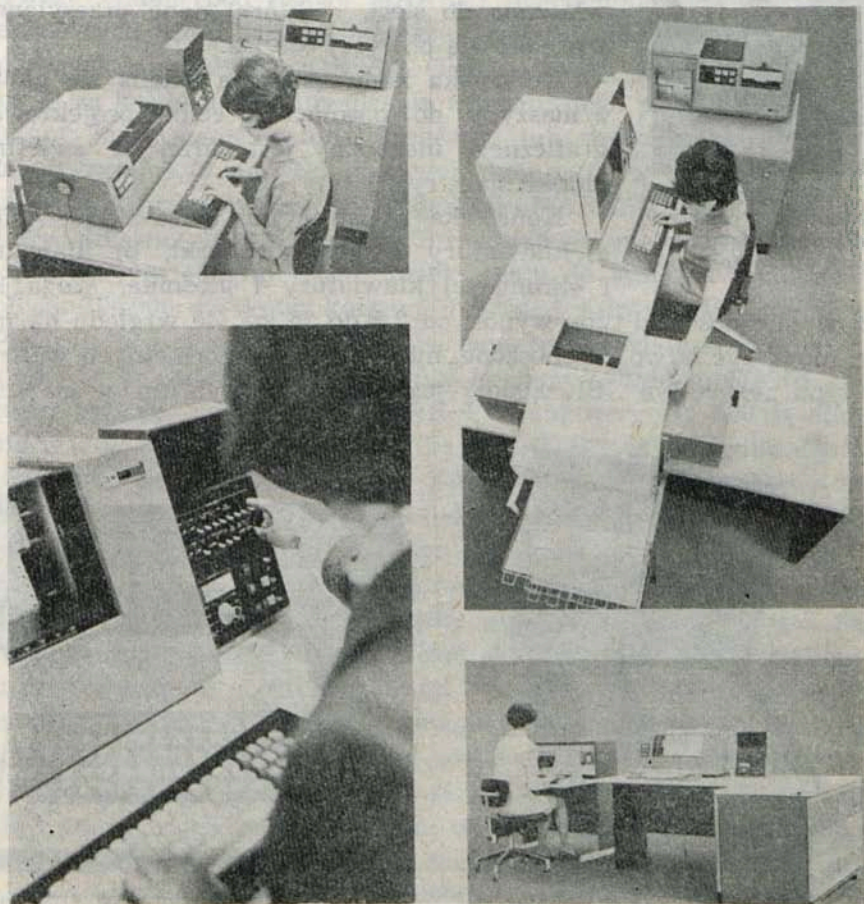
INSERT 200,50

system wypisuje na końcówce

250

a użytkownik wstawia dane

250 PO DESZCZU JEST SŁOŃCE



Ryc. 2.24. Końcówka przetwarzania typu IBM 2770