

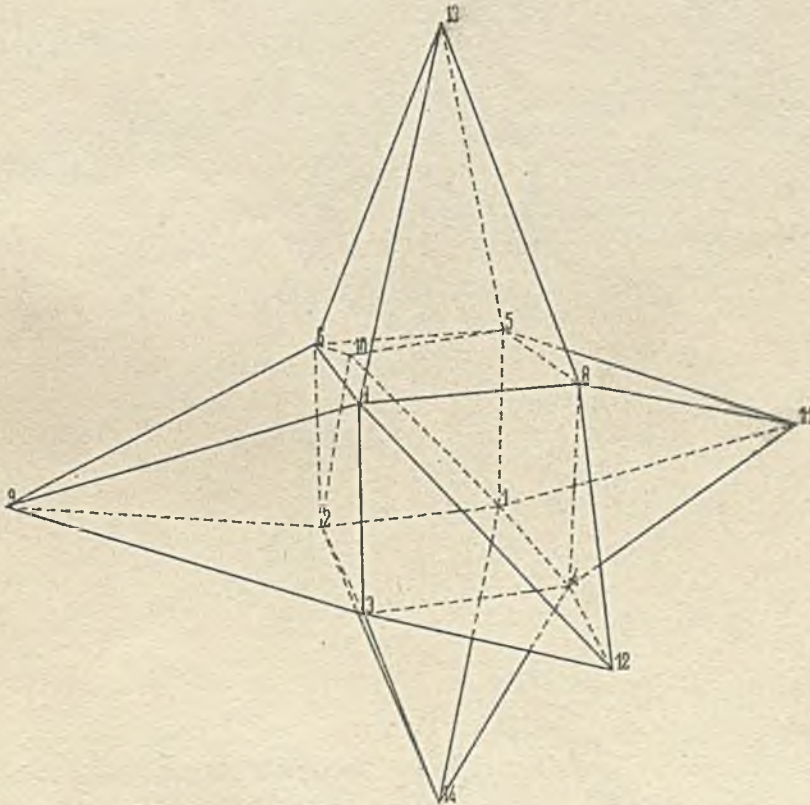
P. 3057 / 82

biuletyn informacyjny

3-4
'82



NAUKI
I TECHNIKI
KOMPUTEROWE



Rysunek na okładce:

Odzworowanie perspektywiczne podobiektu
GWIAZDA. Zob.opracowanie: Wąsowska A.,
Wąsowski M. na stronie 23.



P. 3057/82

Rok XX

Nr 3-4

1982

Streszczenie

Содержание

Contents

BERTHOLD A., TOPOLSKI S.: Uwagi na temat zastosowania ploterów - ze szczególnym uw- zględnieniem zestawu kreślącego XYNETICS 1200 Hewlett Packard - zainstalowanego w Instytucie Maszyn Matematycznychs. 3	БЕРТОЛЬД А., ТОПОЛЬСКИ С.: Замечания по теме применения плотеров с учётом чертёжного комплекта XYNETICS 1200 Hewlett Packard - установленного в Ин- ституте Математических Машинс. 3	BERTHOLD A., TOPOLSKI S.: Remarks on the plotter applica- tion including the XYNETICS 1200 Hewlett Packard plotting set installed in the Institute of Mathematical Machines.....p. 3
WĄSOWSKA A., WĄSOWSKI M.: Kompu- terowe kreślenie rysunków pers- pektywicznych obiektów przestrzennychs.23	ВОНСОВСКА А., ВОНСОВСКИ М.: Компьютерное черчение перспекти- вных чертежей пространственных объектовс.23	WĄSOWSKA A., WĄSOWSKI M.: Computer perspective drawing of space objects.....p.23
KACPRZAK M.: Sieć lokalna ETHERNET.....s.31	КАЦПЖАК М.: Местная сеть ETHERNETс.31	KACPRZAK M.: The ETHERNET local netp.31
BONKOWICZ-SITTAUER S., GRODZICKI S.: Oprogramowanie biblioteczne minikomputera MERA 400s.41	КАЦПЖАК М.: Местная сеть ETHERNETс.31	BONKOWICZ-SITTAUER S., GRODZICKI S.: MERA 400 minicom- puter library software.....p.41
PAWLIK R.: Język do przetwarza- nia danych statystycznych w bibliotece fortranowskiej systemu MERA 400s.47	БОНКОВИЧ-СИТТАУЭР С., ГРОДЗИЦКИ С.: Библиотечное про- грамное обеспечение мини-ЭВМ MERA 400с.41	PAWLIK R.: Language for the statistic data processing in the FORTRAN library of the MERA 400 systemp.47
WASIL G.: Zwięzłe listowanie zbiorów źródłowychs.59	ПАВЛИК Р.: Язык для преобразова- ния статистических данных...с.47	WASIL G.: Brief listing of the source filesp.59
Sprawozdania z konferencjis.65	ВАСИЛЬ Г.: Сокращённый листинг исходных файловс.59	Conference reportsp.65
Ofertys.66	Отчёт из конференциис.65	Offersp.66
	Предложенияс.66	

D W U M I E S I E C Z N I K

Wydaje:

I N S T Y T U T M A S Z Y N M A T E M A T Y C Z N Y C H
Branżowy Ośrodek Informacji Naukowej Technicznej i Ekonomicznej

Komitet Redakcyjny

dr inż. Stanisława BONKOWICZ-SITTAUER, doc.dr Jan BOROWIEC,
mgr Cezary DZIADOSZ /sekretarz redakcji/
doc. dr inż. Jan ŁYSKANOWSKI, doc. dr hab.inż.Stanisław MAJERSKI,
doc. dr inż. Henryk ORŁOWSKI /redaktor naczelny/,
dr inż. Piotr PERKOWSKI

Opracowanie redakcyjne: mgr Hanna Drozdowska

Opracowanie graficzne: Barbara Kostrzewska

Adres redakcji: BOINTE IMM, ul.Krzywickiego 34,
02-078 Warszawa, tel. 28-37-29

Biuletyn Informacyjny NAUKI I TECHNIKI KOMPUTEROWE

mgr inż. Adam BERTHOLD

mgr inż. Sławomir TOPOLSKI

Instytut Maszyn Matematycznych

Uwagi na temat zastosowania ploterów

ze szczególnym uwzględnieniem zestawu kreslącego XYNETICS 1200 Hewlett Packard zainstalowanego w Instytucie Maszyn Matematycznych

Wstęp

Opracowanie niniejsze ma na celu zainteresowanie czytelników zastosowaniem ploterów (autokreślarek). Dlatego też, oprócz przedstawienia konkretnej konfiguracji sprzętowej działającej w IMM, podane w nim zostaną przykłady zastosowania ploterów w bardzo różnych dziedzinach. Być może uwagi te pozwolą czytającemu określić czy prowadzone przez niego prace można usprawnić stosując graficzne przedstawienie wyników obliczeń.

Zastosowanie ploterów

W ciągu ostatnich kilku lat zauważyć można dynamiczny rozwój systemów grafiki komputerowej. Odnosi się to zarówno do sprzętu, jak i do oprogramowania. Wynika to z jednej strony z rozwoju elektroniki, dzięki której sprzęt graficzny stał się tani i łatwo dostępny, z drugiej zaś wzrost zainteresowania technikami graficznymi jako prostym narzędziem ułatwiającym i skracającym czas wykonywania bardzo różnych prac. Łatwiej jest niektórym rzeczy ocenić, zweryfikować lub wreszcie uzmysłowić sobie na podstawie rysunku lub wykresu niż na podstawie długich kolumn liczb. Produkcję sprzętu podjęło wiele poważnych firm, a jednocześnie powstało dużo różnych systemów graficznych ukierunkowanych czasem na jedną tylko sferę zainteresowań. Liczne wskazówki, a nawet gotowe algorytmy dotyczące projektowania takich systemów można znaleźć w literaturze [1].

Należy zauważyć, że rysunek jako graficzny obraz wyników obliczeń może być albo produktem finalnym, albo też stanowić etap przejściowy przyspieszający osiągnięcie produktu finalnego.

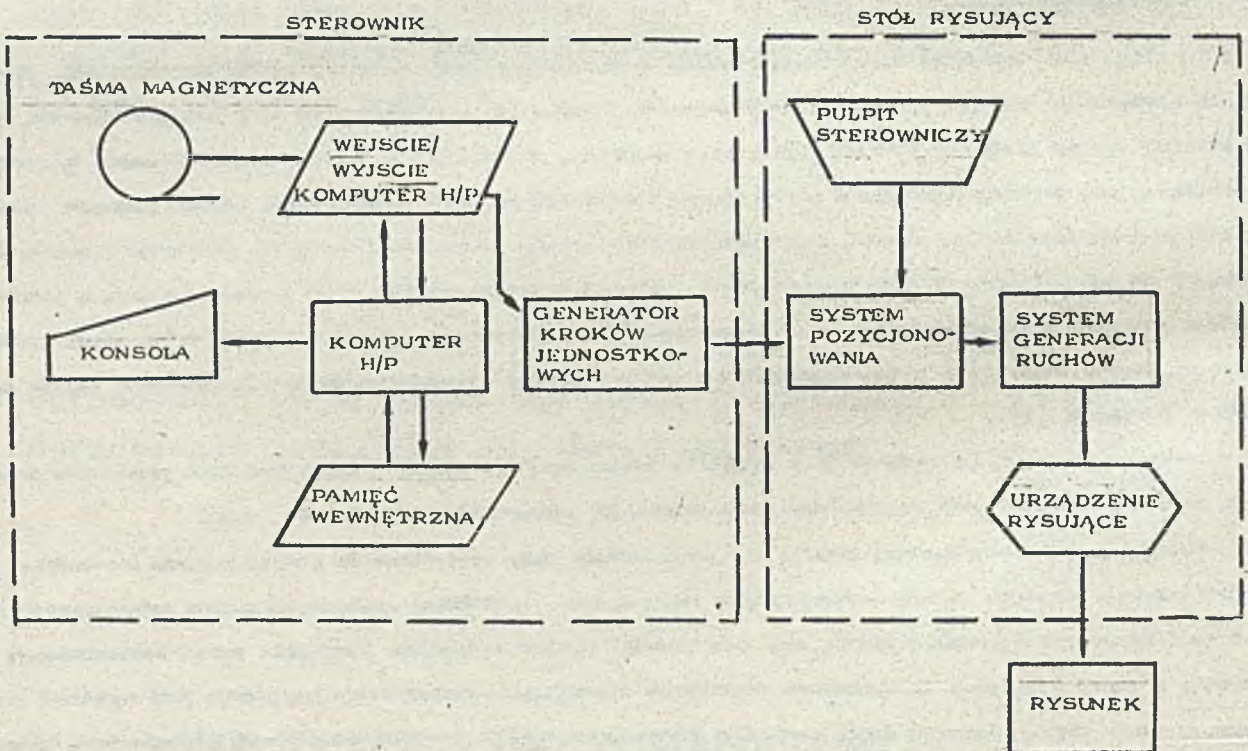
Metody grafiki komputerowej stosuje się powszechnie przy projektowaniu i wykonywaniu elementów części maszyn. Stosując metody numerycznych wspomaganie technikami graficznymi można łatwo zaprojektować wariantowe rozwiązania projektu, aby móc później wybrać optymalne. Następnie przed sprawdzeniem projektu w metalu (np. przy zastosowaniu obrabiarek sterowanych numerycznie) wygodnie jest wykonać najpierw rysunek zaprojektowanej drogi narzędzia skrawającego. Taka metoda weryfikacji jest znacznie tańsza i szybsza od metod warsztatowych. Innego typu zastosowanie to tworzenie dokumentacji, zarówno w potocznym tego słowa znaczeniu, jak również dokumentacji kontrolno-pomiarowej. Z takim zastosowaniem spotykamy się np. w przemyśle lotniczym. Dokumentacja warstwowa (tzw. plazowa), która w chwili obecnej wykonywana jest ręcznie metodą trasowania i przechowywana w postaci arkuszy blachy może być z powodzeniem, po wygenerowaniu kształtu przedmiotu w pamięci komputera, wyrysowana automatycznie za pomocą odpowiedniej autokreślarki np. na astralonie. Ma to tę dodatkową zaletę, że można z niej wtedy bezpośrednio wykonywać dokumentację warsztatową. Metody takie są znane i powszechnie stosowane w produjących firmach lotniczych. Łatwo wyobrazić sobie, jakie usługi może oddać autokreślarka (traktowana jako dodatkowe urządzenie wyjściowe) w takich dyscyplinach jak architektura, urbanistyka czy budownictwo. Podobnie sytuacja

wygląda w kartografii, gdzie intuicyjnie wyczuwamy potrzebę wizualizacji wyników programów, jak również w meteorologii i jej pokrewnych. Ploter może być jednak pomocny również w takiej dziedzinie jak zarządzanie. Nie ma ona pozornie wiele wspólnego z rysowaniem a jednak przedstawiona graficznie sieć działań, analiza czasów, kosztów lub nawet zestawienie statystyczne bardziej przemawia do wyobraźni niż w jakiegokolwiek innej formie.

Przykłady zastosowań tego typu można mnożyć, nie jest to jednak głównym celem niniejszego opracowania. Należy tylko wspomnieć jeszcze o usługach, jakie metody graficzne oddają w pracach naukowo-badawczych, planowaniu i dokumentowaniu eksperymentów naukowych, itp.

Opis zestawu kreślącego działającego w IMM

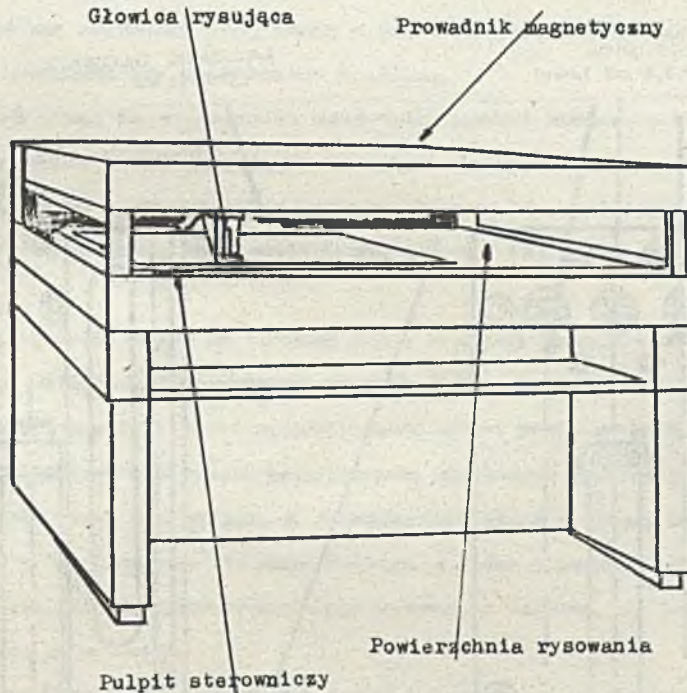
Zestaw kreślący działający w IMM składa się z dwóch podstawowych elementów: stołu rysującego (firmy XYNETICS) oraz sterownika (firmy HEWLETT PACKARD). Zestaw pracuje w trybie off-line w stosunku do bazowej emy, którą jest R-32. Jego zaletą jest duży obszar rysowania, wynoszący około 1470x2200 mm oraz duża dokładność i powtarzalność rysowania ok. 0,1 mm, tan. rzędu grubości rysowanej linii.



Rys.1. Schemat działania systemu ADS

Istotna jest także prędkość rysowania, wynosząca maksymalnie około 8 cm/s. Na rysunku 1 przedstawiony jest schemat działania systemu ADS(Automated Drafting System), będącego standardowym wyposażeniem zestawu. Zadaniem sterownika jest pobranie danych wejściowych, dokonanie na nich niezbędnych działań

przy uwzględnieniu informacji (instrukcji) z konsoli operatora i przesłanie ich w postaci przetworzonej do urządzenia rysującego, pokazanego na rysunku 2. Dane wejściowe przenoszone są do komputera HEWLETT PACKARD z bazowej EMC za pośrednictwem taśmy magnetycznej o standardzie IBM (typu SL, gęstość

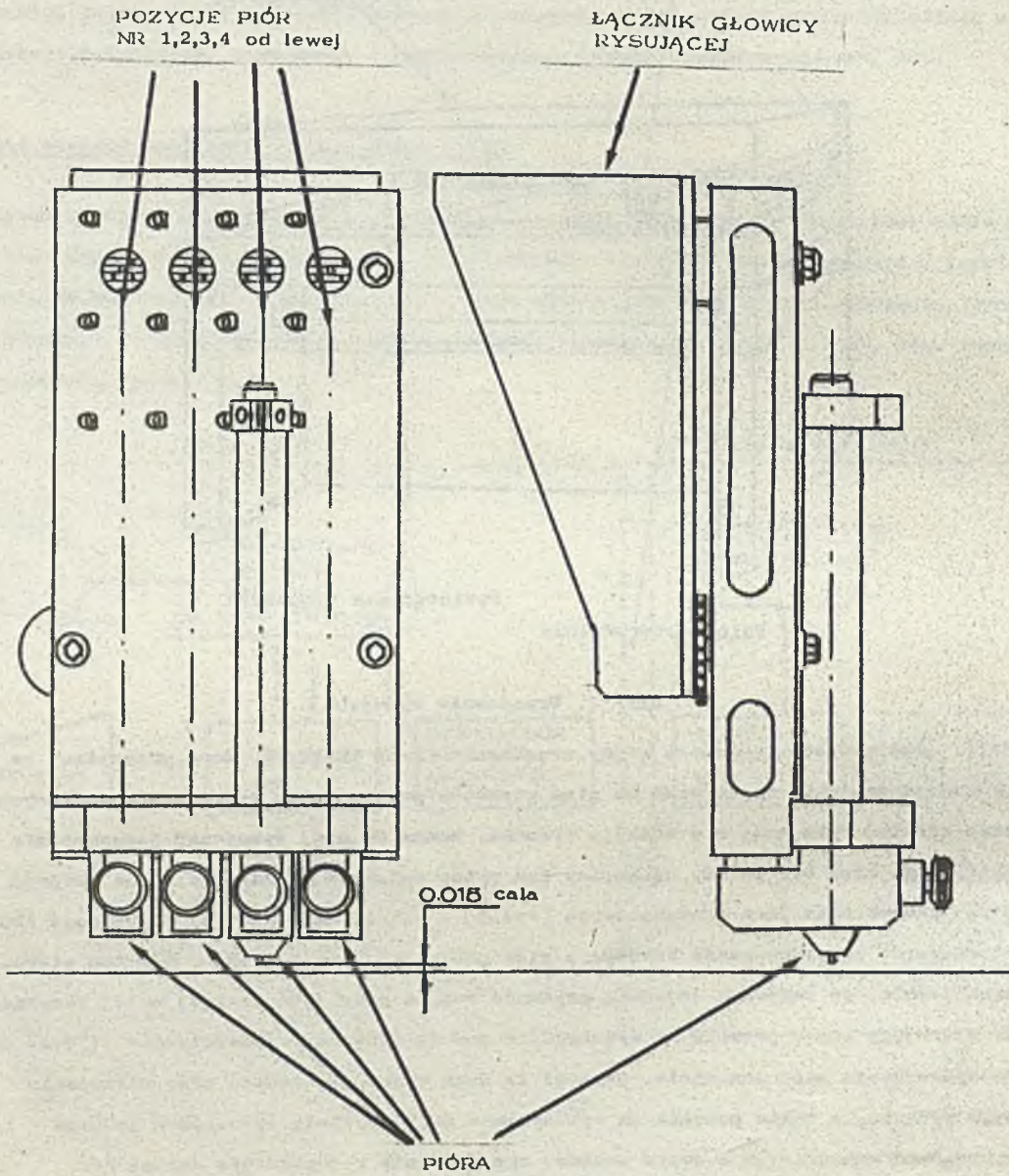


Rys. 2. Urządzenie rysujące

zapieju 800 BPI). Jako konsola operatora służy urządzenie firmy TELETYPE. Dane przesyłane ze sterownika podlegają zamianie w ploterze na ciąg rozkazów powodujących wykonanie określonych czynności przez głowicę rysującą, a w efekcie rysunku. Można do niej zamocować jednocześnie ostery "pisaki". Mogą nimi być pióra, długopisy lub rylce zależnie od materiału, na którym wykonywany jest rysunek oraz jego przeznaczenie (rys. 3). Należy zauważyć, że w systemie ADS następuje automatyczne pozycjonowanie każdego z piór (do pozycji rysującej). Głowica sterowana jest magnetycznie, co zapewnia jej dużą płynność ruchów przy jednoczesnej małej bezwładności. Pulpit sterujący stołu pozwala na wykonywanie ruchów głowicą w płaszczyźnie XY oraz na podnoszenie i opuszczanie piór manualnie. Stanowi to dużą wygodę zwłaszcza przy ustawianiu początku nowego rysunku, a także pozwala na wykonywanie kolejno wielu rysunków z jednego zbioru. Opis urządzeń wchodzących w skład zestawu znajduje się w "Operators manual for XYNETIOS AUTOMATED DRAFTING SYSTEM" [2].

Oprogramowanie

Oprogramowanie zestawu kreślącego składa się z dwóch "części" - jednej działającej na sterowniku HEWLETT PACKARD i drugiej stanowiącej oprogramowanie bazowej EMC tzn. R-32. System ADS stanowiący



Rys.3. Schemat głowicy rysującej

oryginalne oprogramowanie sterownika nadzoruje cały proces rysowania łącznie z wczytaniem danych wejściowych. Pozwala on użytkownikowi na wykonywanie (za pomocą instrukcji napisanych na konsoli) pewnych manipulacji głowicą rysującą takich, jak przesuwanie jej do dowolnego punktu, wracanie do początku układu bazowego, podnoszenie i opuszczanie pióra. Z konsoli można określić nowy układ współrzędnych, podziałkę, prędkość rysowania, numer pióra rysującego, gabaryty rysunku, itp. Również za pomocą komend podawanych na konsoli dokonuje się czynności związanych z działaniem przewijaka taśmy magnetycznej, jak np. ustawianie jej w zadane położenie lub odwijanie do początku.

System ADS pozwala także na wypisywanie zawartości pamięci komputera i jej zmienianie, usuwanie i dopisywanie specjalistycznych modułów do systemu, ustawianie punktów zatrzymania oraz wykonywanie pewnych czynności związanych z przesuwaniem papieru. Na konsolę operatora system wypisuje komunikaty o wykrytych błędach, których dokładny opis znajduje się w poz. [2]. Tam też znajdują się bliższe dane o działaniu systemu.

Druga część oprogramowania działa na bazowej EMC. W chwili obecnej ukończone są prace nad postprocesorem o nazwie PLOTER, pozwalającym na wyrysowywanie drogi narzędzia zaprogramowanej w systemie NARVIK. Pewne uwagi na temat programowania w tym systemie zawarte są w poz. [3]. Schematycznie "miejsce" plotera w procesie projektowania obrabiarzek sterowanych numerycznie (do czego służy system NARVIK) pokazane jest na rys. 4. Jednocześnie widać z niego, że rysunek otrzymywany na autokreślarkę nie jest w tym procesie obiektem finalnym, a tylko elementem weryfikacji programu. Weryfikacja ta jest o tyle celowa, że nie tylko przyspiesza wykonanie zadania, ale jest ponadto o wiele tańsza od weryfikacji warsztatowej tj. w metalu.

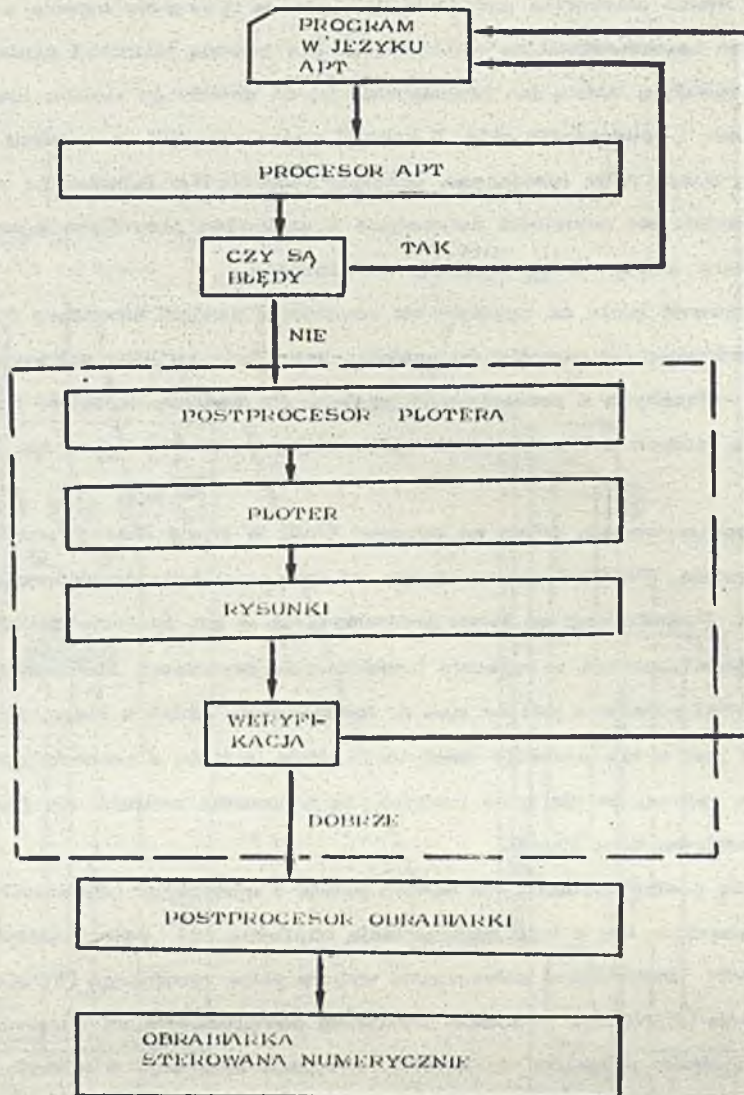
Posługiwanie się postprocesorem jest bardzo proste i wymaga od użytkownika niewielkiego tylko nakładu pracy. Po zapoznaniu się z jego dokumentacją użytkową [4] należy uzupełnić program dla procesora systemu NARVIK instrukcjami dotyczącymi wyboru pióra rysującego (TOOLNO), jego podnoszenia (PENUP) i opuszczania (PENOWN). Pozostałe instrukcje postprocesora są opcjonalne i dotyczą segmentowania rysunku. Na wydruku programu użytkownik otrzymuje informacje o punkcie początkowym rysowania oraz automatycznie wyliczonych gabarytach rysunku. Na ich podstawie sporządza instrukcję dla operatora zestawu.

W wypadku wykrycia w programie błędów postprocesor sygnalizuje ich wystąpienie oraz blokuje nagranie na taśmę magnetyczną danych wyjściowych.

Ukończona jest również biblioteka typowych procedur o nazwie XYNBEN do wykorzystywania w języku FORTRAN IV na EMC R-32. Opracowana ona została na podstawie standardowych bibliotek ploterów XYNETICS i BENSON. Zachowano przy tym używaną w nich formę odwołań do procedur. W ten sposób użytkownik przyzwyczajony do działania na podstawie oryginalnego oprogramowania zachodniego nie musi przyswajać sobie żadnych nowych wiadomości.

Sposób postępowania w celu uzyskania graficznych wyników obliczeń za pomocą biblioteki XYNBEN przedstawi ono poglądowo na rys. 6. Jediną czynnością, jaką musi wykonać użytkownik (poza wywołaniem w kartach sterujących odpowiedniej procedury systemowej) jest uzupełnienie programu źródłowego instrukcjami CALL z odpowiednią nazwą, zależnie od charakteru przeznaczonej do rysowania linii. Opis użytkowy pakietu znajduje się w [5].

Procedury pakietu XYNBEN umożliwiają rysowanie dowolnych funkcji zdefiniowanych matematycznie,



Rys. 4. "Miejsce" plotera w procesie projektowania obrabiarek sterowanych numerycznie

ruchy po liniach łamanych od punktu do punktu, opisywanie rysunku itp. Istnieje również tak jak i w postprocesorze PLOTER możliwość wybierania dowolnych piór oraz ich podnoszenia i opuszczania. Dodatkowym udogodnieniem jest możliwość definiowania różnych krzywych w kilku jednocześnie układach współrzędnych.

Oprócz pakietu XYNBEN przewidziane jest także w przyszłości wykonywanie innych o mniej uniwersalnym przeznaczeniu na indywidualne zlecenia odbiorców.

Przykład programu z użyciem biblioteki BENKYN (fragmenty) :

```

00 1 SUBROUTINE TECK(X,Y,Z)
00 2 A=2*Z
00 3 B=5*Z
00 4 C=4*Z
00 5 CALL TRAA(X,Y,0)
00 6 CALL TRAA(0,0,C/2)
00 7 CALL TRAA(A/2,C/2)
00 8 CALL TRAA(-Z,C/2)
00 9 CALL TRAA(0,0,-B/3)
00 10 CALL TRAA(Z,C/2)
00 11 CALL TRAA(A/2,0/2)
00 12 RETURN
00 13 END

```

OPTIONS IN EFFECT ID,EBCDIC,SOURCE,NOLIST,MODECK,LOAD,NOMAP
OPTIONS IN EFFECT NAME = TE, LINECNT = 50
STATISTICS SOURCE STATEMENTS = 13,PROGRAM SIZE = 608
STATISTICS NO DIAGNOSTICS GENERATED

```

00 1 SUBROUTINE GIE(X,Y,Z)
00 2 A=2*Z
00 3 B=5*Z
00 4 C=3*Z
00 5 CALL TRAA(X,Y,0)
00 6 CALL TRAA(C,C/2)
00 7 CALL TRAA(-Z,C/2)
00 8 CALL TRAA(-Z/2,C/2)
00 9 CALL TRAA(0,0,-A/3)
00 10 CALL TRAA(Z,C/2)
00 11 CALL TRAA(Z,C/2)
00 12 CALL TRAA(0,0,C/2)
00 13 CALL TRAA(0,0,-B/3)
00 14 CALL TRAA(-Z/2,C/2)
00 15 CALL TRAA(-Z,C/2)
00 16 CALL TRAA(-Z,C/2)
00 17 CALL TRAA(-Z,C/2)
00 18 CALL TRAA(-Z,C/2)
00 19 CALL TRAA(B,C/2)
00 20 RETURN
00 21 END

```

OPTIONS IN EFFECT ID,EBCDIC,SOURCE,NOLIST,MODECK,LOAD,NOMAP
OPTIONS IN EFFECT NAME = GIE, LINECNT = 50
STATISTICS SOURCE STATEMENTS = 21,PROGRAM SIZE = 900
STATISTICS NO DIAGNOSTICS GENERATED

```

00 1 SUBROUTINE I(X,Y,Z)
00 2 A=6*Z
00 3 B=3*Z
00 4 C=2*Z
00 5 CALL TRAA(X,Y,C)
00 6 CALL TRAA(0.0,A,Z)
00 7 CALL TRAA(0.0,-Z,Z)
00 8 CALL TRAA(0.0,-Z,Z)
00 9 CALL TRAA(0.0,-B,Z)
00 10 CALL TRAA(Z,-Z,Z)
00 11 CALL TRAA(C,0.0,Z)
00 12 RETURN
00 13 END
JSP 2.0 FORT 90K

```

DATE = 06.12.82 TIME = 14.20.64 PAGE 0002

```

*OPTIONS IN EFFECT* ID=EBCDIC,SOURCE=NOLIST,MODECK,LOAD=UOMAP
*OPTIONS IN EFFECT* NAME = II LIMECHT = 50
*STATISTICS* SOURCE STATEMENTS = 132 PROGRAM SIZE = 618
*STATISTICS* NO DIAGNOSTICS GENERATED
JSP 2.0 FORT 90K

```

DATE = 06.12.82 TIME = 14.20.64 PAGE 0001

```

00 1 REAL*4 AA(3,30), BB(3,30), CC(6,30), T(30)
00 2 CALL DIMTAB(0,1,1)
00 3 ID=18
00 4 DO 300 NRE=1,2
00 5 ALFA=0.10
00 6 DLOS=65.00
00 7 DLDZ= 9.10
00 8 M=1
00 9 LK=0
00 10 Y=-8.5
00 11 W=20.0
00 12 CALL PEN(1)
00 13 DO 200 J=1,2
00 14 CALL ROTATE(ALFA)
00 15 DLOS=DLOS+25.0
00 16 DLDZ=DLDZ- 2.0
00 17 CALL PLOT(0.0,-5.0,13)
00 18 CALL PLOT(0.0,DLOS,12)
00 19 CALL TRAS(-2.0,-4.0,3)
00 20 CALL TRAS(-2.0,-4.0,2)
00 21 CALL TRAS(-2.0,-4.0,3)
00 22 LK=LK+10
00 23 A=0.10
00 24 B=2.0*M
00 25 DO 10 I=1,LK
00 26 V=DLDZ*I
00 27 CALL PLOT(A,V,13)
00 28 CALL PLOT(B,V,12)
00 29 LJ=2
00 30 CW=2*A
00 31 D=2*B
00 32 W=1+15.0
00 33 DO 20 I=1,LJ
00 34 Z=1+I
00 35 CALL PLOT(C,Z,13)
00 36 CALL PLOT(D,Z,12)
00 37 20 ALFA=ALFA-90.0
00 38 CALL ROTATE(0.0)
00 39 X=12.0

```

```

0106 P=P*00L15
0107 X=T(1)
0108 Y=CCCNK(1)*P
0109 CALL PLOT(X,Y,M3)
0110 DO 600 I=241D
0111 X=T(I)
0112 Y=CCCNK(I)*P
0113 600 CALL PLOT(X,Y,I2)
0114 610 CONTINUE
0115 GK=CK
0116 R=PR
0117 KP=L*KP
0118 620 CALL ECHEL(1,70,11,0,0,0,0,0)
C
0119 OPISY SYMBOLARY
0120 CALL PCARS(40,0,0,17,0,0,0,0)
0121 CALL II(45,0,47,0,1,0)
0122 CALL GIE(49,0,17,0,1,0)
0123 CALL PLOT(0,0,0,0,0,5)
0124 300 CONTINUE
0125 CALL PLOT(0,0,0,0,999)
0126 STOP
0127 1000 FORMAT(4F8.5)
0128 1100 FORMAT(8F8.5)
0129 1200 FORMAT(1M)
0130 1300 FORMAT(1H,7(5X,F8.5))
0131 1400 FORMAT(1H,8(5X,F8.5))
0132 1500 FORMAT(F8.5)
END
JSP 240 FORT 90K

```

DATE = 06:12:82 TIME = 14:20 04 PAGE 0004

OPTIONS IN EFFECT ID,EBCDIC,SOURCE,NOLIST,MODECK,LOAD,NOHAP
OPTIONS IN EFFECT NAME = MAIN, LINECT = 50
STATISTICS SOURCE STATEMENTS = 132, PROGRAM SIZE = 5282
STATISTICS NC DIAGNOSTICS GENERATED
STATISTICS NO DIAGNOSTICS THIS STEP

X	10,000	0,0	YE	-0,000	10,000	0,0	YE	-0,000	10,000	0,0	YE	STA=UP	PEN=1	10,000	-0,000
X=	10,000	0,0	YE	-0,000	10,000	0,0	YE	-0,000	10,000	0,0	YE	STA=UP	PEN=1	10,000	-0,000
X=	10,000	0,0	YE	-0,000	10,000	0,0	YE	-0,000	10,000	0,0	YE	STA=DWN	PEN=1	10,000	-2,000
X=	15,000	0,0	YE	-0,000	15,000	0,0	YE	-0,000	15,000	0,0	YE	STA=UP	PEN=1	15,000	-0,000
X=	15,000	0,0	YE	-0,000	15,000	0,0	YE	-0,000	15,000	0,0	YE	STA=DWN	PEN=1	15,000	-2,000
X=	20,000	0,0	YE	-0,000	20,000	0,0	YE	-0,000	20,000	0,0	YE	STA=UP	PEN=1	20,000	-0,000
X=	20,000	0,0	YE	-0,000	20,000	0,0	YE	-0,000	20,000	0,0	YE	STA=DWN	PEN=1	20,000	-2,000
X=	25,000	0,0	YE	-0,000	25,000	0,0	YE	-0,000	25,000	0,0	YE	STA=UP	PEN=1	25,000	-0,000
X=	25,000	0,0	YE	-0,000	25,000	0,0	YE	-0,000	25,000	0,0	YE	STA=DWN	PEN=1	25,000	-2,000
X=	30,000	0,0	YE	-0,000	30,000	0,0	YE	-0,000	30,000	0,0	YE	STA=UP	PEN=1	30,000	-0,000
X=	30,000	0,0	YE	-0,000	30,000	0,0	YE	-0,000	30,000	0,0	YE	STA=DWN	PEN=1	30,000	-2,000
X=	35,000	0,0	YE	-0,000	35,000	0,0	YE	-0,000	35,000	0,0	YE	STA=UP	PEN=1	35,000	-0,000
X=	35,000	0,0	YE	-0,000	35,000	0,0	YE	-0,000	35,000	0,0	YE	STA=DWN	PEN=1	35,000	-2,000
X=	40,000	0,0	YE	-0,000	40,000	0,0	YE	-0,000	40,000	0,0	YE	STA=UP	PEN=1	40,000	-0,000
X=	40,000	0,0	YE	-0,000	40,000	0,0	YE	-0,000	40,000	0,0	YE	STA=DWN	PEN=1	40,000	-2,000
X=	45,000	0,0	YE	-0,000	45,000	0,0	YE	-0,000	45,000	0,0	YE	STA=UP	PEN=1	45,000	-0,000
X=	45,000	0,0	YE	-0,000	45,000	0,0	YE	-0,000	45,000	0,0	YE	STA=DWN	PEN=1	45,000	-2,000
X=	50,000	0,0	YE	-0,000	50,000	0,0	YE	-0,000	50,000	0,0	YE	STA=UP	PEN=1	50,000	-0,000
X=	50,000	0,0	YE	-0,000	50,000	0,0	YE	-0,000	50,000	0,0	YE	STA=DWN	PEN=1	50,000	-2,000
X=	55,000	0,0	YE	-0,000	55,000	0,0	YE	-0,000	55,000	0,0	YE	STA=UP	PEN=1	55,000	-0,000
X=	55,000	0,0	YE	-0,000	55,000	0,0	YE	-0,000	55,000	0,0	YE	STA=DWN	PEN=1	55,000	-2,000
X=	60,000	0,0	YE	-0,000	60,000	0,0	YE	-0,000	60,000	0,0	YE	STA=UP	PEN=1	60,000	-0,000
X=	60,000	0,0	YE	-0,000	60,000	0,0	YE	-0,000	60,000	0,0	YE	STA=DWN	PEN=1	60,000	-2,000
X=	65,000	0,0	YE	-0,000	65,000	0,0	YE	-0,000	65,000	0,0	YE	STA=UP	PEN=1	65,000	-0,000
X=	65,000	0,0	YE	-0,000	65,000	0,0	YE	-0,000	65,000	0,0	YE	STA=DWN	PEN=1	65,000	-2,000
X=	70,000	0,0	YE	-0,000	70,000	0,0	YE	-0,000	70,000	0,0	YE	STA=UP	PEN=1	70,000	-0,000
X=	70,000	0,0	YE	-0,000	70,000	0,0	YE	-0,000	70,000	0,0	YE	STA=DWN	PEN=1	70,000	-2,000
X=	75,000	0,0	YE	-0,000	75,000	0,0	YE	-0,000	75,000	0,0	YE	STA=UP	PEN=1	75,000	-0,000
X=	75,000	0,0	YE	-0,000	75,000	0,0	YE	-0,000	75,000	0,0	YE	STA=DWN	PEN=1	75,000	-2,000
X=	80,000	0,0	YE	-0,000	80,000	0,0	YE	-0,000	80,000	0,0	YE	STA=UP	PEN=1	80,000	-0,000
X=	80,000	0,0	YE	-0,000	80,000	0,0	YE	-0,000	80,000	0,0	YE	STA=DWN	PEN=1	80,000	-2,000
X=	85,000	0,0	YE	-0,000	85,000	0,0	YE	-0,000	85,000	0,0	YE	STA=UP	PEN=1	85,000	-0,000
X=	85,000	0,0	YE	-0,000	85,000	0,0	YE	-0,000	85,000	0,0	YE	STA=DWN	PEN=1	85,000	-2,000
X=	90,000	0,0	YE	-0,000	90,000	0,0	YE	-0,000	90,000	0,0	YE	STA=UP	PEN=1	90,000	-0,000
X=	90,000	0,0	YE	-0,000	90,000	0,0	YE	-0,000	90,000	0,0	YE	STA=DWN	PEN=1	90,000	-2,000
X=	95,000	0,0	YE	-0,000	95,000	0,0	YE	-0,000	95,000	0,0	YE	STA=UP	PEN=1	95,000	-0,000
X=	95,000	0,0	YE	-0,000	95,000	0,0	YE	-0,000	95,000	0,0	YE	STA=DWN	PEN=1	95,000	-2,000
X=	100,000	0,0	YE	-0,000	100,000	0,0	YE	-0,000	100,000	0,0	YE	STA=UP	PEN=1	100,000	-0,000
X=	100,000	0,0	YE	-0,000	100,000	0,0	YE	-0,000	100,000	0,0	YE	STA=DWN	PEN=1	100,000	-2,000
X=	50,000	0,0	YE	-0,000	50,000	0,0	YE	-0,000	50,000	0,0	YE	STA=UP	PEN=1	50,000	-0,000
X=	50,000	0,0	YE	-0,000	50,000	0,0	YE	-0,000	50,000	0,0	YE	STA=DWN	PEN=1	50,000	-2,000
X=	100,000	0,0	YE	-0,000	100,000	0,0	YE	-0,000	100,000	0,0	YE	STA=UP	PEN=1	100,000	-0,000
X=	100,000	0,0	YE	-0,000	100,000	0,0	YE	-0,000	100,000	0,0	YE	STA=DWN	PEN=1	100,000	-2,000

NOWY UKLAD LOKALNY X= 0,0															
SX= 1,000															
SY= 1,000															
ALFA= 0															

RYSOJANIE ZNAKOK	0,5	0,5	YE	33,667	33,667	0,0	YE	33,667	33,667	0,0	YE	STA=DWN	PEN=1	-6,000	33,667
X=	-6,000	0,0	YE	33,667	33,667	0,0	YE	33,667	33,667	0,0	YE	STA=DWN	PEN=1	-6,000	33,667
RYSOJANIE ZNAKOK	1,0	1,0	YE	68,667	68,667	0,0	YE	68,667	68,667	0,0	YE	STA=DWN	PEN=1	-7,333	68,667
X=	-7,333	0,0	YE	68,667	68,667	0,0	YE	68,667	68,667	0,0	YE	STA=DWN	PEN=1	-7,333	68,667
RYSOJANIE ZNAKOK	1,0	1,0	YE	-7,833	-7,833	0,0	YE	-7,833	-7,833	0,0	YE	STA=DWN	PEN=1	51,667	-7,833
X=	51,667	0,0	YE	-7,833	-7,833	0,0	YE	-7,833	-7,833	0,0	YE	STA=DWN	PEN=1	51,667	-7,833
RYSOJANIE ZNAKOK	2,0	2,0	YE	-7,833	-7,833	0,0	YE	-7,833	-7,833	0,0	YE	STA=DWN	PEN=1	101,000	-7,833
X=	101,000	0,0	YE	-7,833	-7,833	0,0	YE	-7,833	-7,833	0,0	YE	STA=DWN	PEN=1	101,000	-7,833
X=	111,000	0,0	YE	8,800	8,800	0,0	YE	8,800	8,800	0,0	YE	STA=UP	PEN=1	111,000	8,800
X=	113,400	0,0	YE	8,800	8,800	0,0	YE	8,800	8,800	0,0	YE	STA=DWN	PEN=1	113,400	8,800
X=	112,200	0,0	YE	11,200	11,200	0,0	YE	11,200	11,200	0,0	YE	STA=UP	PEN=1	112,200	11,200
X=	112,200	0,0	YE	5,200	5,200	0,0	YE	5,200	5,200	0,0	YE	STA=DWN	PEN=1	112,200	5,200
X=	113,400	0,0	YE	4,000	4,000	0,0	YE	4,000	4,000	0,0	YE	STA=DWN	PEN=1	113,400	4,000
X=	5,263	0,0	YE	35,000	35,000	0,0	YE	35,000	35,000	0,0	YE	STA=UP	PEN=1	3,263	35,000
X=	94,737	0,0	YE	35,000	35,000	0,0	YE	35,000	35,000	0,0	YE	STA=DWN	PEN=1	94,737	35,000

X#	78,947	(11,579	MM	69,761	(0,997	MM	PEN=1	STA=DWH	(78,947,	69,761)
X#	84,210	(11,684	MM	71,912	(1,027	MM	PEN=1	STA=DWH	(84,210,	71,912)
X#	89,473	(11,789	MM	73,436	(1,049	MM	PEN=1	STA=DWH	(89,473,	73,436)
X#	94,737	(11,895	MM	74,489	(1,064	MM	PEN=1	STA=DWH	(94,737,	74,489)
X#	5,263	(01,105	MM	5,263	(0,707	MM	PEN=1	STA=UP	(5,263,	5,263)
X#	10,526	(01,211	MM	5,319	(0,707	MM	PEN=1	STA=DWH	(10,526,	5,319)
X#	15,789	(01,310	MM	5,433	(0,707	MM	PEN=1	STA=DWH	(15,789,	5,433)
X#	21,052	(01,421	MM	5,673	(0,707	MM	PEN=1	STA=DWH	(21,052,	5,673)
X#	26,316	(01,520	MM	6,174	(0,707	MM	PEN=1	STA=DWH	(26,316,	6,174)
X#	31,579	(01,632	MM	7,259	(0,714	MM	PEN=1	STA=DWH	(31,579,	7,259)
X#	36,842	(01,737	MM	9,760	(0,739	MM	PEN=1	STA=DWH	(36,842,	9,760)
X#	42,105	(01,842	MM	15,888	(0,227	MM	PEN=1	STA=DWH	(42,105,	15,888)
X#	47,368	(01,947	MM	29,981	(0,428	MM	PEN=1	STA=DWH	(47,368,	29,981)
X#	52,631	(11,053	MM	50,519	(0,723	MM	PEN=1	STA=DWH	(52,631,	50,519)
X#	57,894	(11,158	MM	64,612	(0,923	MM	PEN=1	STA=DWH	(57,894,	64,612)
X#	63,158	(11,263	MM	70,740	(1,011	MM	PEN=1	STA=DWH	(63,158,	70,740)
X#	68,421	(11,368	MM	73,241	(1,046	MM	PEN=1	STA=DWH	(68,421,	73,241)
X#	73,684	(11,474	MM	74,326	(1,062	MM	PEN=1	STA=DWH	(73,684,	74,326)
X#	78,947	(11,579	MM	74,827	(1,069	MM	PEN=1	STA=DWH	(78,947,	74,827)
X#	84,210	(11,684	MM	75,067	(1,072	MM	PEN=1	STA=DWH	(84,210,	75,067)
X#	89,473	(11,789	MM	75,181	(1,074	MM	PEN=1	STA=DWH	(89,473,	75,181)
X#	94,737	(11,895	MM	75,232	(1,075	MM	PEN=1	STA=DWH	(94,737,	75,232)

ALFAM 0.0

0.10 1,000

0.0 1,000

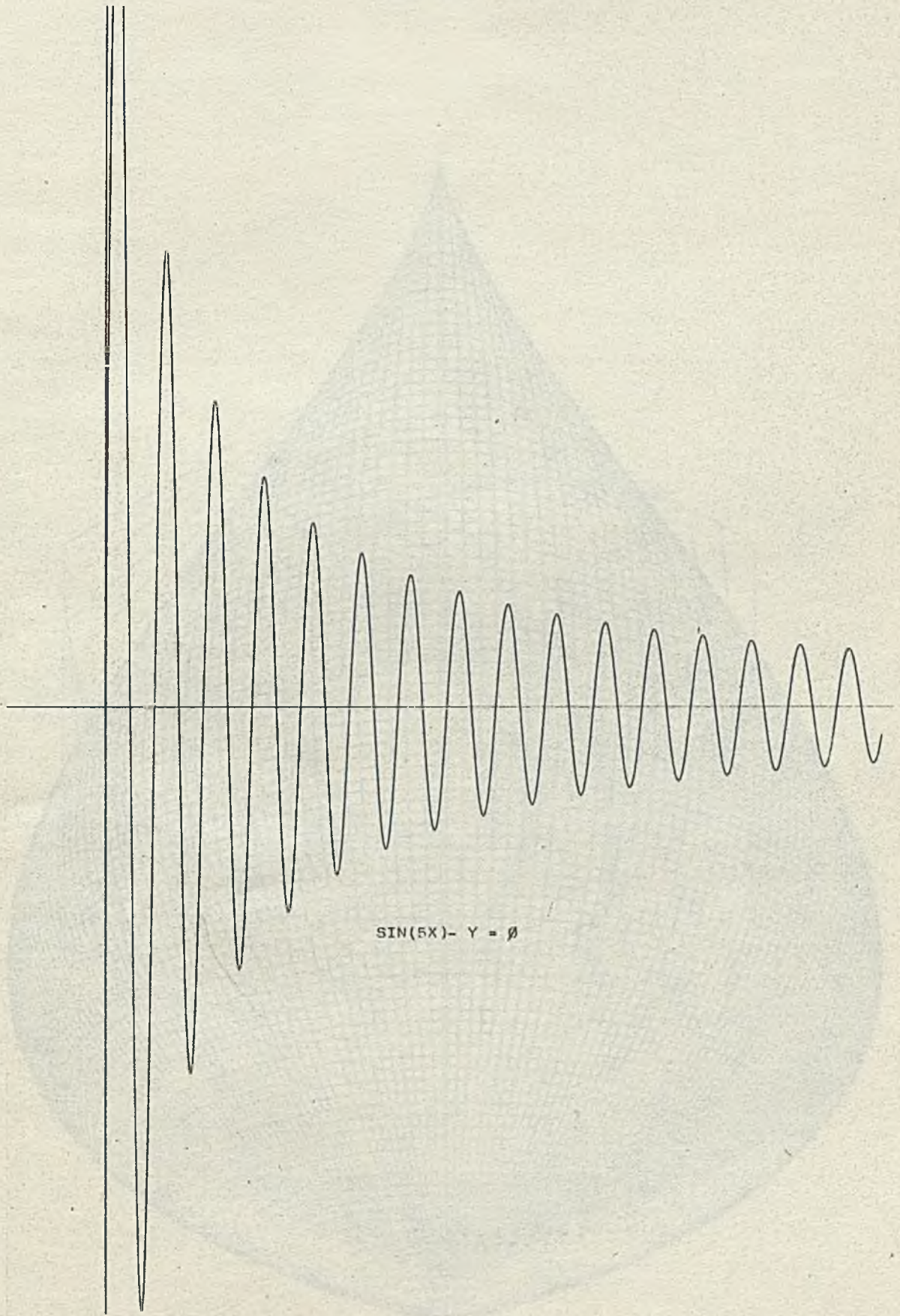
NOUJ UKLAD LOKALNY

RYSOVANIE ZNAKOV

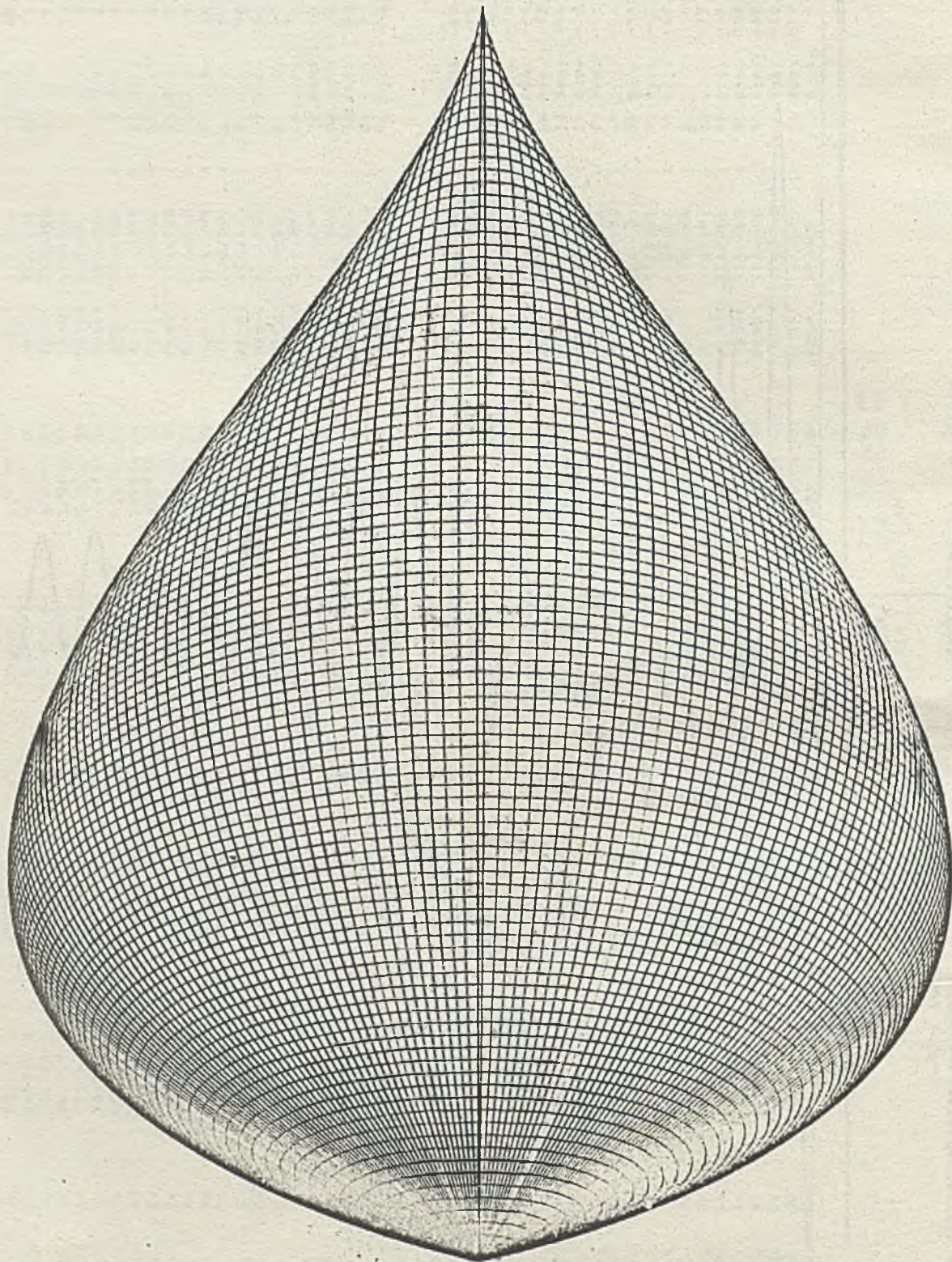
X#	73,333	(73,333	MM	-17,000	(-17,000	MM	PEN=1	STA=DWH	(73,333,	-17,000)
X#	45,000	(45,000	MM	-11,000	(-11,000	MM	PEN=1	STA=UP	(45,000,	-11,000)
X#	45,000	(45,000	MM	-12,000	(-12,000	MM	PEN=1	STA=DWH	(45,000,	-12,000)
X#	45,000	(45,000	MM	-13,000	(-13,000	MM	PEN=1	STA=UP	(45,000,	-13,000)
X#	45,000	(45,000	MM	-16,000	(-16,000	MM	PEN=1	STA=DWH	(45,000,	-16,000)
X#	46,000	(46,000	MM	-17,000	(-17,000	MM	PEN=1	STA=DWH	(46,000,	-17,000)
X#	51,000	(51,000	MM	-14,000	(-14,000	MM	PEN=1	STA=UP	(51,000,	-14,000)
X#	50,000	(50,000	MM	-13,000	(-13,000	MM	PEN=1	STA=DWH	(50,000,	-13,000)
X#	49,000	(49,000	MM	-14,000	(-14,000	MM	PEN=1	STA=DWH	(49,000,	-14,000)
X#	49,000	(49,000	MM	-16,000	(-16,000	MM	PEN=1	STA=DWH	(49,000,	-16,000)
X#	51,000	(51,000	MM	-17,000	(-17,000	MM	PEN=1	STA=DWH	(51,000,	-17,000)
X#	52,000	(52,000	MM	-16,000	(-16,000	MM	PEN=1	STA=DWH	(52,000,	-16,000)
X#	52,000	(52,000	MM	-13,000	(-13,000	MM	PEN=1	STA=UP	(52,000,	-13,000)
X#	52,000	(52,000	MM	-18,000	(-18,000	MM	PEN=1	STA=DWH	(52,000,	-18,000)
X#	51,000	(51,000	MM	-19,000	(-19,000	MM	PEN=1	STA=DWH	(51,000,	-19,000)
X#	49,000	(49,000	MM	-19,000	(-19,000	MM	PEN=1	STA=DWH	(49,000,	-19,000)
X#	54,000	(54,000	MM	-18,000	(-18,000	MM	PEN=1	STA=DWH	(54,000,	-18,000)
X#	0,0	(0,0	MM	-17,000	(-17,000	MM	PEN=1	STA=UP	(0,0,	-17,000)

GABARYTY RYSUNKU
 XMI= -12,000 MM DX= 127,000 MM
 YMI= -19,000 MM DY= 199,000 MM
 0.0 MM 0.70 MM

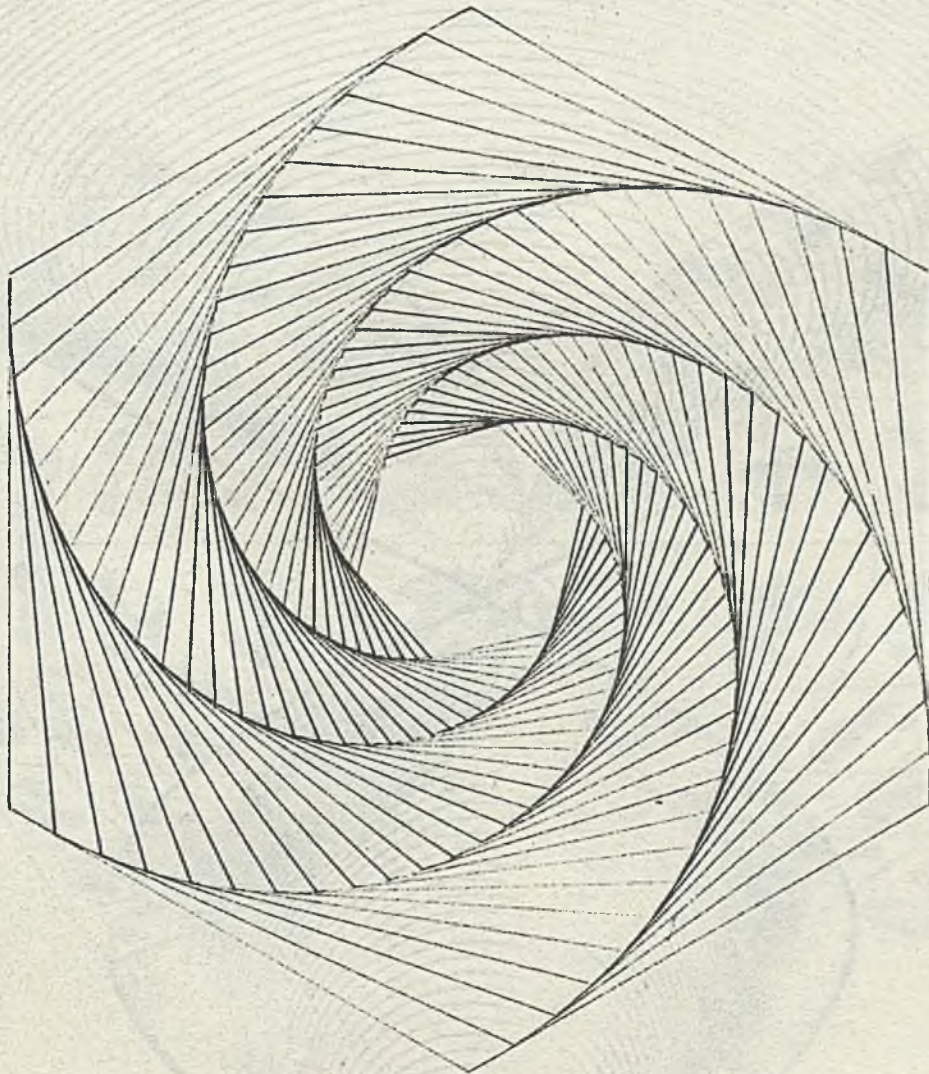
NUMERY LOGICZNE UZYWANYCH PIOR - i 3
 & BLOKOW ZACZYNA SIE OD 1 BLOKU



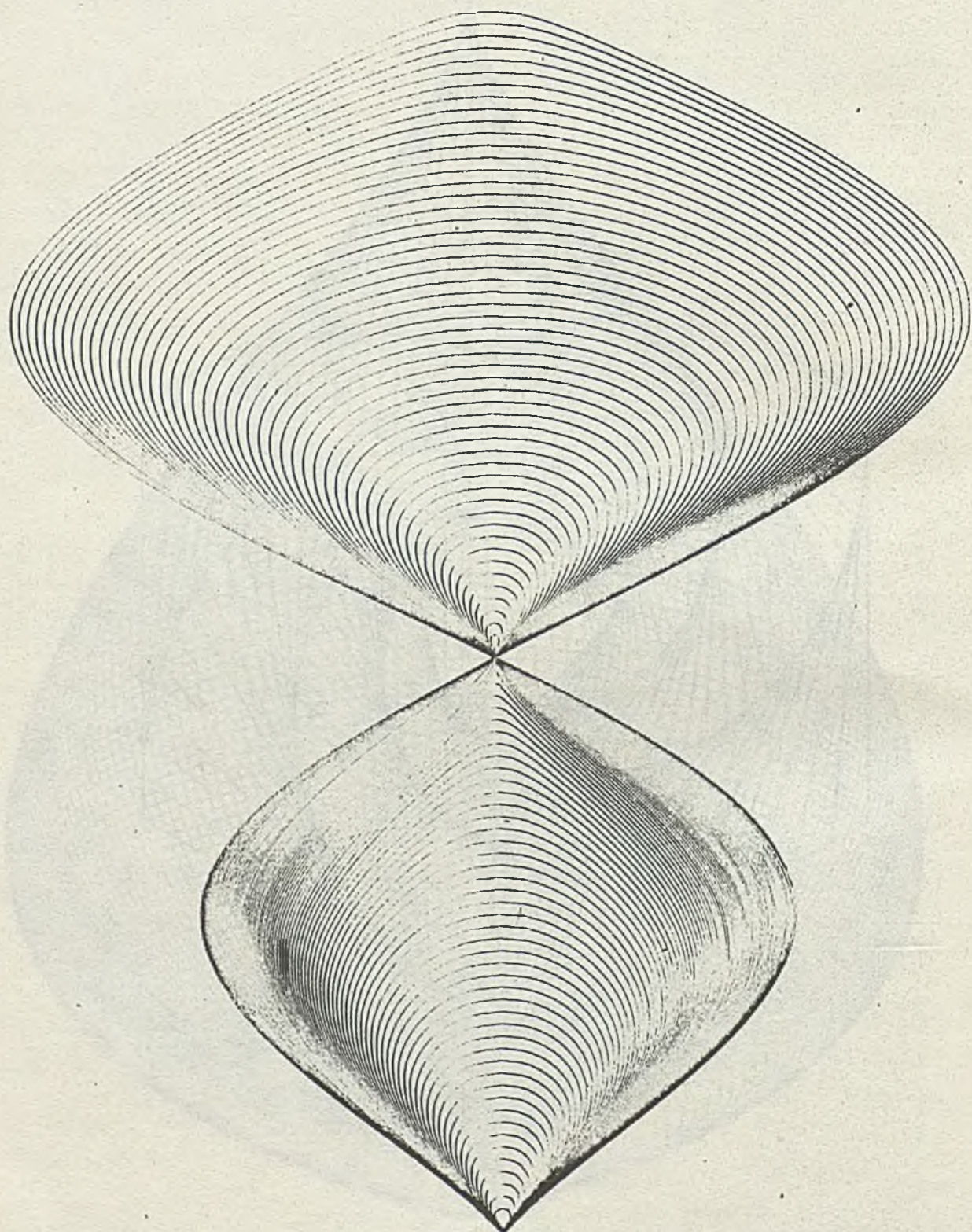
Rys. 5 a-f). Przykłady rysunków wykonanych za pomocą biblioteki XYNBEN

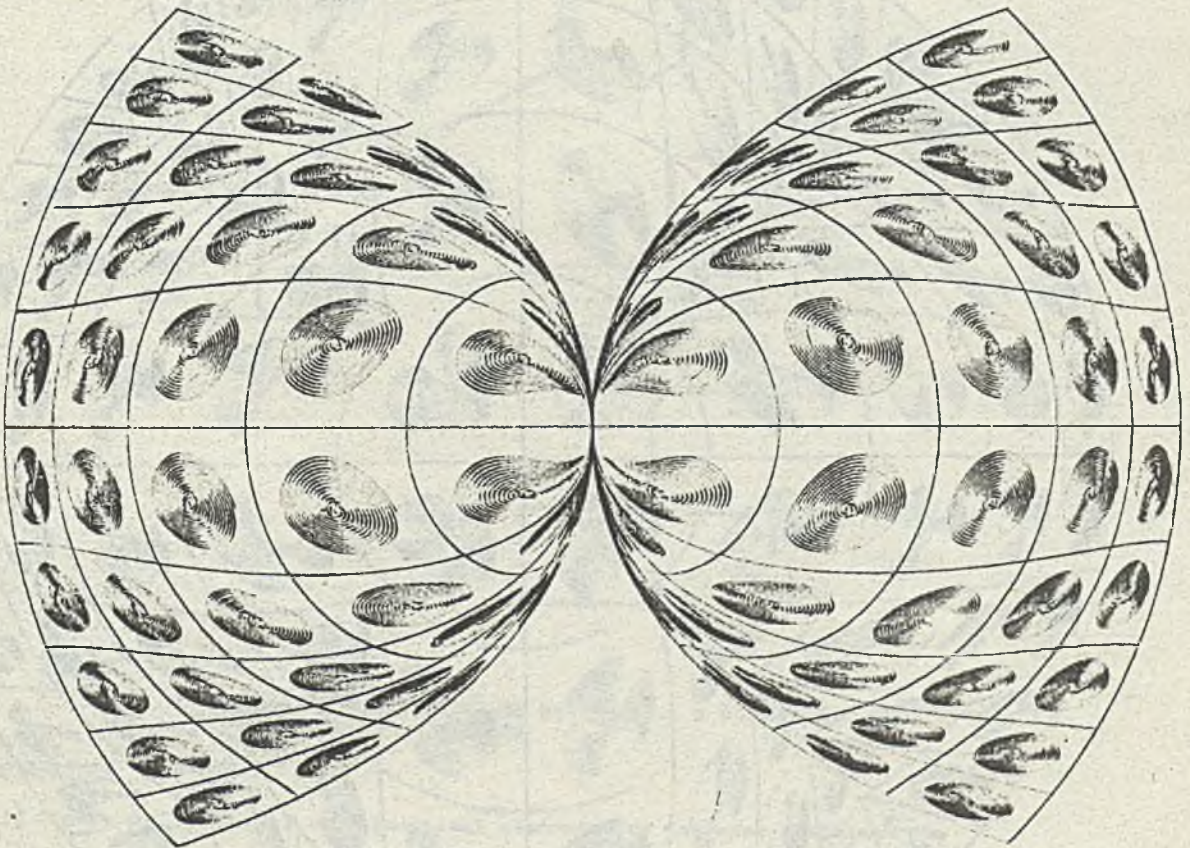


Rys. 5 b)



Rys. 5 c)





Rys. 5 e)

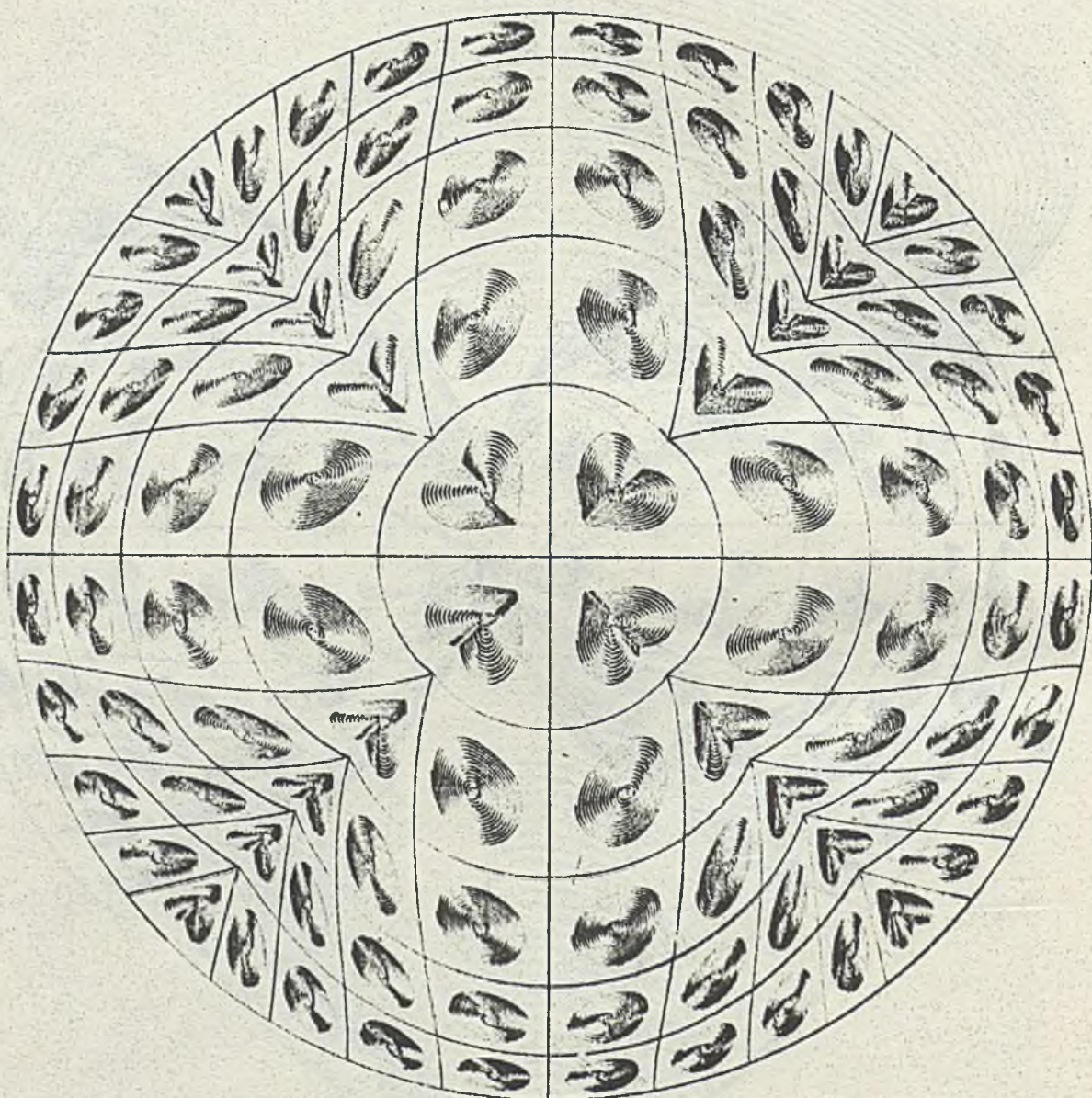
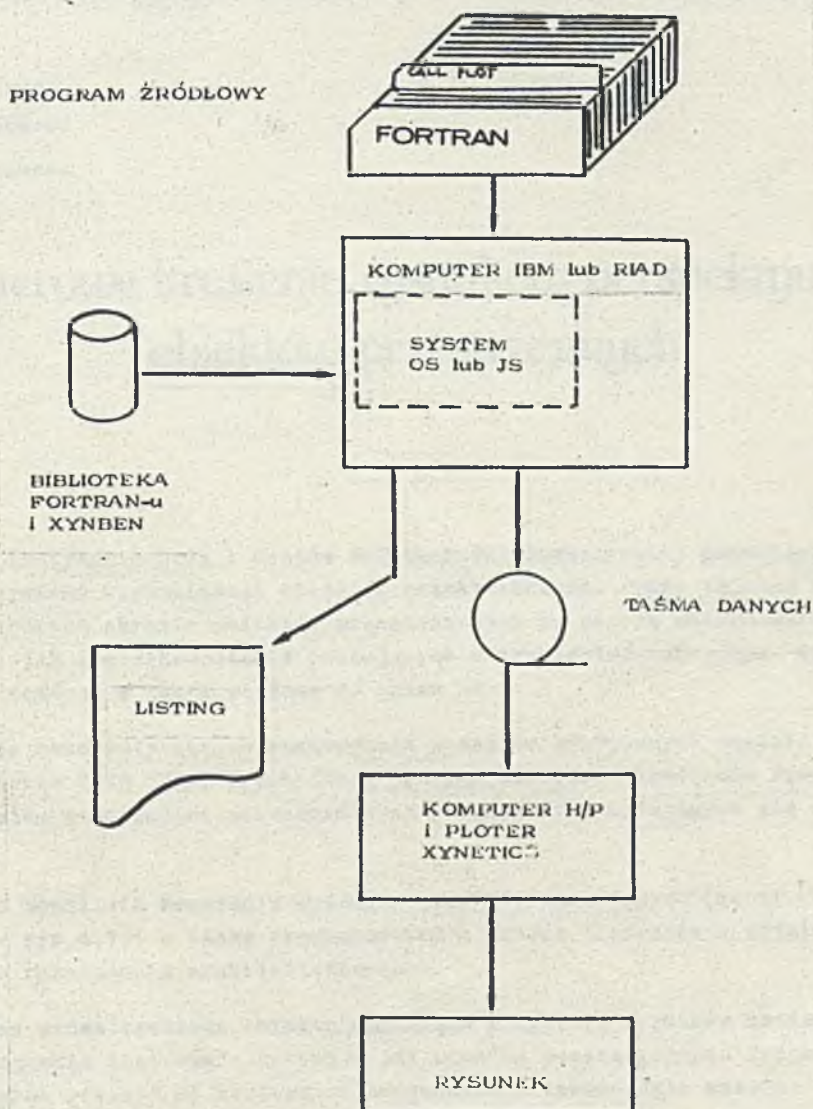


Рис. 5 f)



Rys. 6. Sposób korzystania z biblioteki XYNBEN

Literatura

- [1] Rogers D., Adams J.: - Mathematical elements for Computer Graphics
- [2] Operators manual for XYNETICS AUTOMATED DRAFTING SYSTEM
- [3] Berthold A., Galicki J., Topolski S.: Automatyczna synteza elementów programu dla obrabia-
rek sterowanych numerycznie. BIOSK 1979 nr 1
- [4] Berthold A., Galicki J., Lewkowiec J., Topolski S.: Opis użytkowy postprocesora PLOTER /masz./
- [5] Berthold A., Galicki J., Lewkowiec J., Topolski S.: Opis użytkowy pakietu
podstawowych funkcji kreślących BENXTN... /maszynopis/

Biuletyn Informacyjny NAUKI I TECHNIKI KOMPUTEROWE

mgr inż. Anna WĄSOWSKA
mgr inż. Marek WĄSOWSKI
Politechnika Warszawska

Komputerowe kreślenie rysunków perspektywicznych obiektów przestrzennych

Wstęp

Od paru lat w Instytucie Dróg i Mostów Politechniki Warszawskiej prowadzone są prace badawcze dotyczące budowy systemu wizualizacji obiektów przestrzennych. Prace te mają na celu umożliwienie tworzenie dwuwymiarowych obrazów obiektów przestrzennych za pomocą maszyn cyfrowych pracujących w trybie wsadowym, jak i minikomputerów pracujących w trybie interakcyjnym. Obydwa te rodzaje systemów wyposażone zostaną w język służący do opisu brył.

Pierwsza wersja prac dotyczących opracowania procedur graficznych została zakończona i uruchomiona na komputerze ODRA 1305. Uruchomiony program umożliwia kreślenie rysunków perspektywicznych z uwzględnieniem widoczności poszczególnych fragmentów znajdujących się w polu widzenia obserwatora.

Oprogramowanie umożliwia kreślenie rysunków obiektów budowlanych (mosty, osiedla, pojedyncze domy, porty itp. - rys. 6, 7), a także wkomponowywanie nowych elementów w istniejący pejzaż i znajdowanie właściwego rozwiązania architektonicznego.

Specyfika opisu geometrycznego obiektu umożliwia kreślenie rysunków zestawieniowych kilku podobieństw (w programie testowym - trzech), jak również poszczególnych fragmentów, można więc wykorzystywać program również do zastosowań związanych z technologią maszyn.

Analogiczne prace, z uwzględnieniem doświadczeń zdobytych podczas pracy na komputerze ODRA 1305, prowadzone są na minikomputerze MERA 400.

Rozpoznawanie widoczności poszczególnych elementów obiektów

Ogólny podział obejmuje trzy klasy algorytmów:

- algorytmy opierające się na badaniu przestrzeni obiektu,
- algorytmy opierające się na badaniu przestrzeni obrazu, obiektu na płaszczyźnie rzutowej,
- algorytmy działające na podstawie listy priorytetów, działające w jednej i drugiej przestrzeni.

Algorytmy należące do pierwszej z wymienionych klas wyznaczają widoczność praktycznie z dokładnością dostępną podczas obliczeń na danym typie komputera, są jednak bardzo kosztowne, gdyż czas obliczeń rośnie w zależności od stopnia komplikacji obiektu. Stosowane są głównie do wykonywania rysunków na autokreślarkach.

W tej grupie można wydzielić dwa podstawowe rodzaje algorytmów:

- algorytm Roberta (1963), który opiera się na sprawdzeniu czy daną krawędź przysłania objętość zajmowana przez jakiś obiekt oraz
- algorytmy analizujące widoczność krawędzi względem innych krawędzi (Appel - 1967,

Loutrel - 1967, Galimberti, Montanari - 1969).

Algorytm Roberta będący najprostszym rozwiązaniem problemu wyznaczania widoczności stosowany może być jedynie do brył wypukłych i ma znaczenie głównie historyczne.

Algorytmy drugiego typu - korzystając z przestrzeni obrazu - są stosowane w wypadku użycia monitora graficznego i dostosowane do specyfiki tego typu urządzenia. Dokładność jest tu uzależniona od wielkości rastra, a czas obliczeń ograniczony ze względu na stałą liczbę elementów rastra (Warnock - 1968, Rommey - 1967, Bouknight - 1969, Watkins - 1970).

Algorytmy korzystające z listy priorytetów, również dostosowane są do zestawów współpracujących z monitorami graficznymi. Różnią się jednak od poprzedniej grupy mniejszą liczbą obliczeń, gdyż elementom obrazu nadawane są priorytety automatycznie (Newell - 1972 - sortowanie ścian pod względem odległości od obserwatora), bądź też z udziałem użytkownika (Schumacker - 1969). Obydwa te algorytmy zostały opracowane w celu tworzenia obrazów w czasie rzeczywistym, przy czym algorytm Schumackera wykorzystuje specjalne urządzenie.

Opis przyjętego rozwiązania

Przyjęte rozwiązanie jest koncepcyjnie zbliżone do rozwiązania Newella, z tym, że zamiast listą priorytetów posługuje się tablicą przecięć ścian. W pierwszej fazie następuje sortowanie ścian w zależności od odległości od obserwatora, co przyspiesza działanie programu. Następnie wyznaczane są punkty przecięć między krawędziami ścian podobiektów na płaszczyźnie rzutowej. W celu usprawnienia działania algorytmu wyznaczane są punkty przecięć między krawędziami w tym samym podobieckcie, a następnie między różnymi podobiektami, jeżeli takie mogą mieć punkty wspólne na płaszczyźnie rzutowej.

Tak więc każda krawędź zostaje podzielona tymi punktami przecięć na odcinki. Następnie ma miejsce sprawdzenie, czy dany odcinek jest przysłaniany przez którąś ze ścian. Dla stwierdzenia, że odcinek jest niewidoczny, wystarczy znalezienie jednej ściany przysłaniającej go. Jeżeli takiej ściany nie znajdziemy - stwierdzamy, że odcinek jest widoczny.

Działanie opracowanego w IDiM PW programu rozpoznawania widoczności elementów obiektu

Uruhomiony program na komputerze ODRA 1305, współpracującym z autokreślarką BENSON jest programem doświadczalnym, mającym służyć do przetestowania niektórych koncepcji, które miały znaleźć zastosowanie przy tworzeniu użytkowego oprogramowania minikomputera MERA 400.

Za pomocą programu można uzyskać rzuty perspektywiczne obiektu złożonego z trzech podobiektów, przy dowolnym położeniu punktu obserwatora. Podział obiektu na podobiektki ma na celu ułatwienie definiowania brył, gdyż każdy z podobiektów jest definiowany w swoim układzie lokalnym.

Opis geometryczny każdego podobiektki składa się:

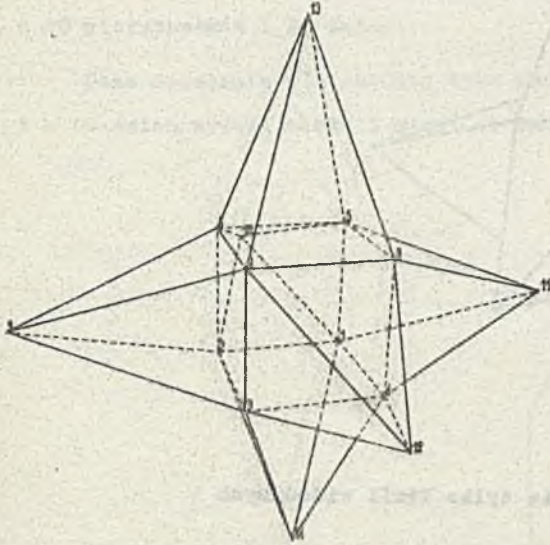
- z wartości współrzędnych przestrzennych wierzchołków i
- z opisu ścian w postaci ukierunkowanego grafu połączeń

Poza tym definiowana jest perspektywa, przez podanie współrzędnych punktu obserwatora oraz punktu głównego co określa kierunek patrzenia. Kąt widzenia podawany jest w stopniach, w zakresie od 0° do 180° , przy przekroczeniu tego zakresu przyjmowany jest standardowo kąt 120° . Przez użytkownika definiowane są również wymiary rysunku, które podawane są w centymetrach.

Po wczytaniu danych oraz upakowaniu ich do odpowiednich tablic, następuje wyznaczenie obszaru znajdującego się w stożku widzenia i odrzucenie fragmentów obiektu znajdujących się poza stożkiem. W dalszej fazie przetwarzane są już tylko dane dotyczące części obiektu znajdującej się w polu widzenia. Przetwarzanie to polega na dokonaniu przekształcenia perspektywicznego, a następnie rozpoznaniu linii widocznych i niewidocznych. Program pozwala na wyrysowanie bądź samych linii widocznych, bądź linii widocznych wraz z niewidocznymi, zaznaczonymi linią kreskową. Użytkownik może również zażądać aby wierzchołki poszczególnych podobiektów były numerowane zgod-

nie z numeracją podaną przy opisie geometrycznym, co ułatwia poprawienie danych w wypadku wystąpienia omyłki. Przykłady rysunku prostego obiektu np. gwiazdy (rys. 1 i 2) oraz obiektu złożonego z dwóch podobiektów (ostrosłup i pudełko bez przykrycia - rys. 3 i 4), ilustrują wykorzystanie tych opcji.

Program ma strukturę nakładkową i wymaga około 32 k słów pamięci operacyjnej.



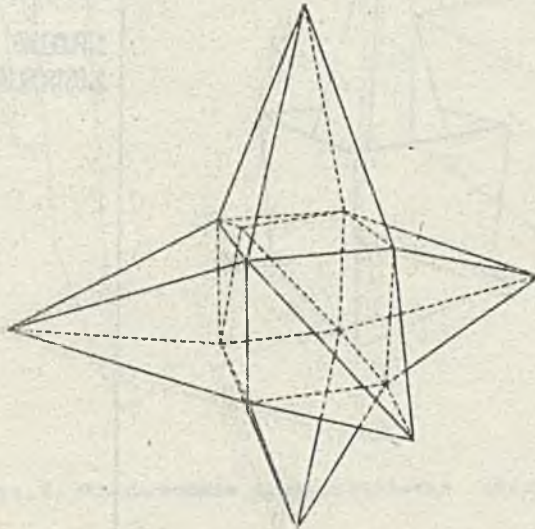
Odzworowanie perspektywiczne podobiektu
GWIAZDA

Współrzędne punktu obserwatora 3, 10, 4

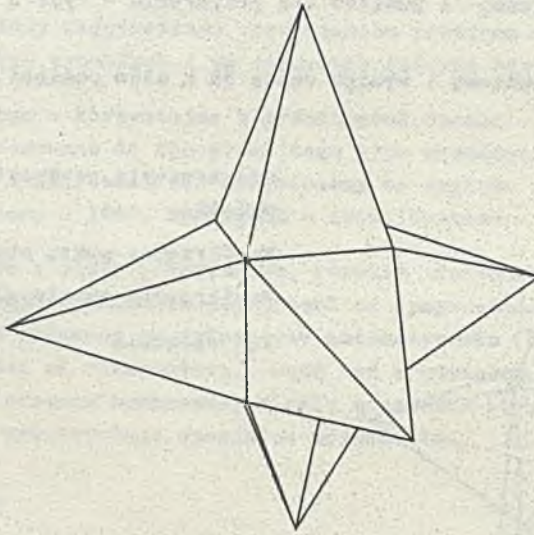
Współrzędne punktu głównego 0, 0, 0

Kąt widzenia 90°

Rys. 1. Zastosowanie opcji kreślenia linii niewidocznych linią przerywaną oraz numerowania wierzchołków



Rys. 2. Zastosowanie opcji kreślenia linii niewidocznych linią przerywaną



Rys. 3. Zastosowanie opcji kreślenia tylko linii widocznych

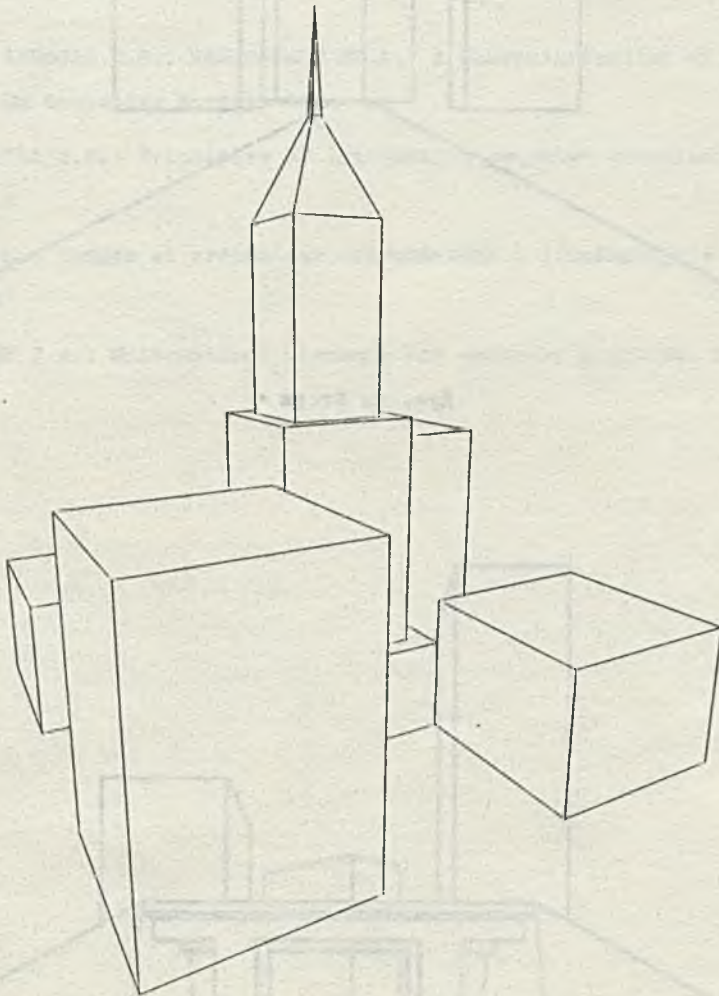
	NR. RYS. 1		
	DATA 16/11/82		
	GODZ. 12/25/10		
	NAZWA OBIEKTU: TEST1		
	NAZWY PODOBIEKTÓW: 1. PUDELKO 2. OSTROSLUP		
PUNKT OBSERWATORA	2000.000	1500.000	1500.000
PUNKT GŁÓWNY	0.000	0.000	500.000
KĄT WIDZENIA	60.000		

Rys. 4. Odzworowanie perspektywiczne obiektu złożonego z dwóch podobiektów - PUDELKO i OSTROSLUP. Rysunek zawiera standardową ramkę z opisem

Ograniczenia programu

Ponieważ przedstawiony program jest jedynie pewnym etapem prac i ma charakter doświadczalny - jego możliwości siłą rzeczy są ograniczone. Ograniczona jest także wielkość podobiektów, z tego względu, że wszystkie zbiory tworzone są w pamięci operacyjnej. Każdy podobiekt może być złożony z 40 wierzchołków i 20 ścian.

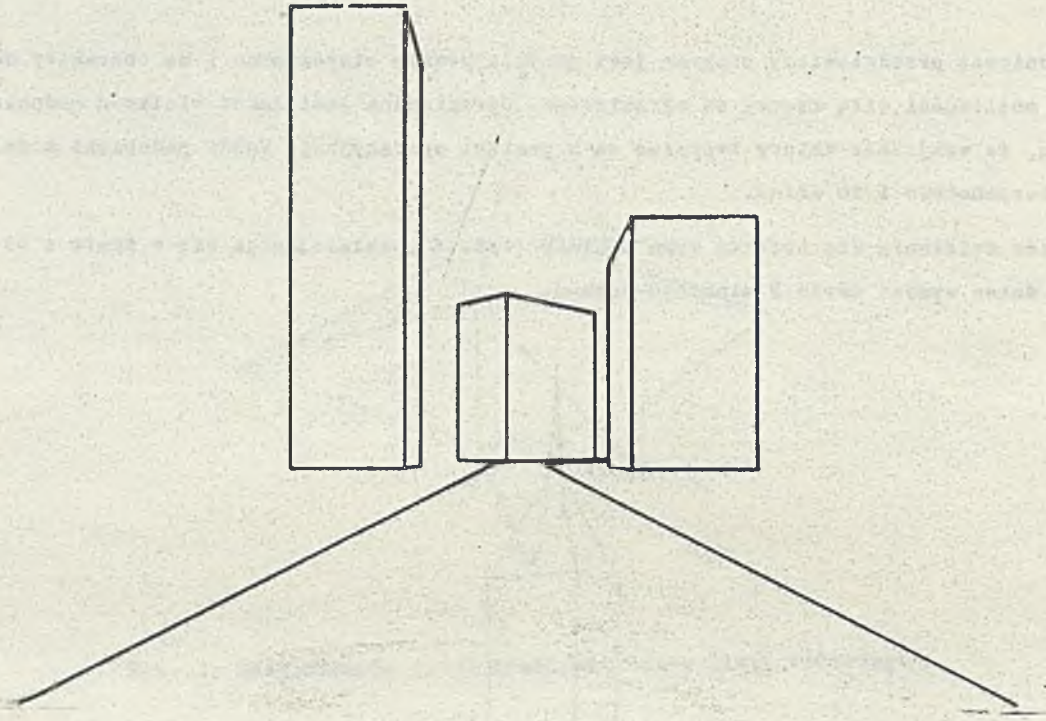
Czas działania dla obiektu typu CENTRUM (rys. 5), składającego się w sumie z 59 wierzchołków i z 45 ścian wynosi około 3 minut 40 sekund.



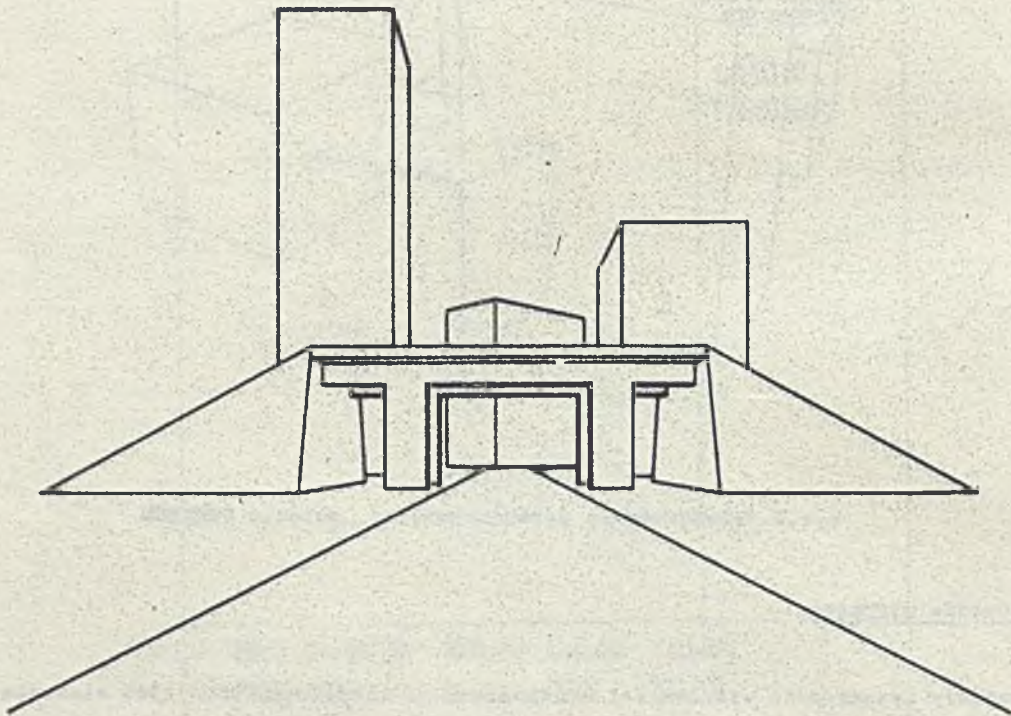
Rys.5. Odwzorowanie perspektywiczne obiektu CENTRUM

Oprogramowanie użytkowe

Procedury wyznaczania widoczności uruchomione na komputerze ODRA 1305 stanowią punkt wyjścia dla stworzenia oprogramowania użytkowego mającego zastosowanie praktyczne wymienione we wstępie. Przyjęto założenie, że oprogramowanie użytkowe ma umożliwić kreślenie rysunków obiektów średniej wielkości około 1000 ścian na minikomputerach MERA 400 w trybie



Rys. 6. Droga -



Rys. 7. Droga z wkomponowanym nowo projektowanym wiaduktem

interakcyjnym. Interakcyjna wersja programu pozwoli na tworzenie obiektu złożonego z opisanych wcześniej podobiektów i nowe wprowadzonych, jego szybką modyfikację oraz bezpośrednią manipulację uzyskiwanym obrazem i obserwację zmian tego obrazu. Przewiduje się utworzenie banku danych opisu geometrycznego elementów najczęściej wykorzystywanych przy tworzeniu obiektów złożonych. W chwili obecnej zakończono zostały prace dotyczące wprowadzania danych oraz procedur umożliwiających uzyskiwanie rysunków perspektywicznych. Obecnie prowadzone są prace dotyczące usprawnienia konwersacji użytkownika z komputerem oraz ułatwienia manipulacji obiektami.

Literatura

- [1] SUTHERLAND I.E., SPROULL R.F., SCHUMACKER BR.A.: A characterization of the Hidden - Surface Algorithms. ACM Computing Surveys 1974 t.6
- [2] NEWMAN W.M., SPROULL R.F.: Principles of interactive computer graphics. Mc Graw Hill, Kogakusha Ltd 1973
- [3] MORVAN P., LUCAS M.: Images et ordinateur introduction à l'infographie interactive. Librairie Larousse 1976
- [4] ROGERS D.F., ADAMS J.A.: Mathematical elements for computer graphics. Mc Graw Hill Book Company 1976

Sieć lokalna ETHERNET

Wstęp

W ramach problemu węzłowego 06.1 "Rozwój techniki komputerowej oraz urządzeń automatyki i aparatury pomiarowej" w Instytucie Maszyn Matematycznych prowadzone były w 1982 r., na silecnie Instytutu Badań Jądrowych prace o charakterze studialnym pt. "System CAMAC w sieciach lokalnych".

Celem prac było przeanalizowanie istniejących rozwiązań i kierunków prac normalizacyjnych w dziedzinie komputerowych sieci lokalnych oraz określenie, które rozwiązanie będzie najkorzystniejsze przy tworzeniu sieci lokalnych na bazie systemu CAMAC. W niniejszym opracowaniu opisano typowy przykład sieci lokalnej, tj. sieć Ethernet, produkowaną obecnie przy współudziale trzech firm: Digital, Intel i Xerox.

Komputerową siecią lokalną nazywa się sieć komputerową, w której maksymalna odległość między dwoma urządzeniami należącymi do sieci wynosi od 15 m do 10 km. Sieć lokalna transmisji danych jest rozmieszczona na niewielkim obszarze, najczęściej w jednym budynku lub w zespole przyległych budynków. Sieć lokalna korzysta z własnych linii komunikacyjnych i należy zazwyczaj do jednej instytucji. Sieć lokalna jest rozwiązaniem pośrednim między długodystansowymi sieciami telekomunikacyjnymi i systemami wielomaszynowymi.

Jedną z pierwszych sieci lokalnych była sieć Ethernet opracowana w 1975 r. przez firmę Xerox. Pierwotną dziedziną zastosowań tej sieci była automatyzacja prac biurowych: wzajemne połączenie urządzeń zbierania, przetwarzania, przechowywania i wyprowadzania tekstów, obrazów cyfrowych lub innych danych oraz umożliwienie szybkiego przesyłania dużych ilości informacji między tymi urządzeniami. Sieć taka zwiększa efektywność wykorzystania urządzeń należących do sieci i upraszcza przesyłanie informacji między różnymi użytkownikami.

W niniejszej pracy opisano wersję Ethernet na podstawie literatury [1], [3]. Obecnie sieci Ethernet - rozwijane i ulepszone w trakcie badań eksperymentalnych - należą do najbardziej rozpowszechnionych sieci lokalnych (zainstalowano ponad 100 takich sieci). Zakres zastosowań Ethernet jest bardzo szeroki: obok automatyzacji prac biurowych, bankowych i handlowych, stosuje się ją do tworzenia dużych systemów przetwarzania danych (obliczeniowych), sterowania procesami i innych zastosowań czasu rzeczywistego (np. zbierania danych laboratoryjnych, nadzorowania pracy urządzeń pomiarowych).

Wobec przyjętej bardzo dużej szybkości przesyłania danych (10 Mb/s), Ethernet może być także stosowana przy tworzeniu systemów wielomaszynowych.

Zasady budowy sieci Ethernet

Przy opracowywaniu założeń na Ethernet za podstawowe wymaganie przyjęto prostotę i niską cenę realizacji sieci. Sieć miała umożliwiać łatwe dołączanie urządzeń przetwarzania danych różnych producentów.

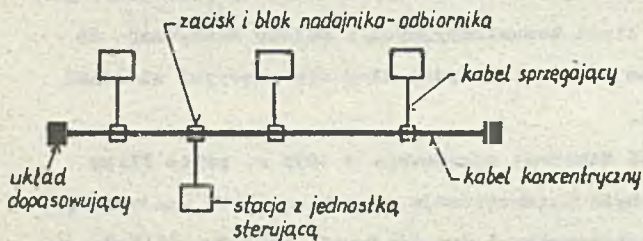
Ponadto ustalono następujące wymagania funkcjonalne:

- architektura sieci powinna umożliwiać bezpośrednią wymianę danych między dwoma urządzeniami, korzystającymi z sieci (bez pośrednictwa trzeciego urządzenia typu "zapamiętaj i wyślij" - store-and-forward),
- powinna istnieć możliwość łatwej rekonfiguracji i rozbudowy sieci (w ramach przyjętych ograniczeń) tak, aby zapewnić proste dołączenie lub odłączenie urządzeń od sieci,
- błąd w jednym urządzeniu dołączonym do sieci nie powinien powodować więcej niż jednego chwilowego błędu w całej sieci,
- sieć powinna mieć cechy, które ułatwią jej konserwację, diagnostykę i obsługę,
- należy zapewnić wielopoziomową, hierarchiczną strukturę protokołów komunikacyjnych,
- sieć powinna być stabilna (odporna na przeciążenia) w tym sensie, że wraz ze wzrostem obciążenia magistrali transmisyjnej, jej wykorzystanie nie powinno maleć.

Architektura sieci

Struktura połączeń elementów sieci

Ethernet składa się z jednego lub kilku tzw. segmentów. Każdy segment (rys.1) ma topologię

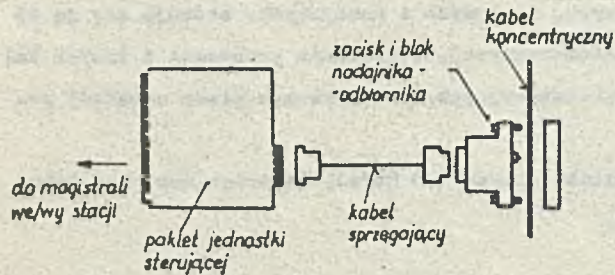


Rys. 1. Segment sieci Ethernet

magistralną (bus). Nośnikiem informacji jest kabel koncentryczny o długości do 500 m, zakończony układami dopasowującymi. Elementy sieci (stacje) są dołączone do kabla koncentrycznego (magistrali) za pomocą zacisku (tap), bloku nadajnika-odbiornika, kabla sprzęgającego i jednostki sterującej. Zastosowanie indukcyjnych sprzężeń z kablem zapewni pełną izolację galwaniczną między magistralą i blokiem nadajnika-odbiornika (blokiem N-O). Dopuszczalna

długość kabla sprzęgającego wynosi 50 m. Do jednego segmentu można dołączyć do 100 bloków N-O (ograniczenie to wynika z przyjętej mocy nadajników i czułości odbiorników, przy uwzględnieniu tłumienności wybranego kabla o przyjętej długości).

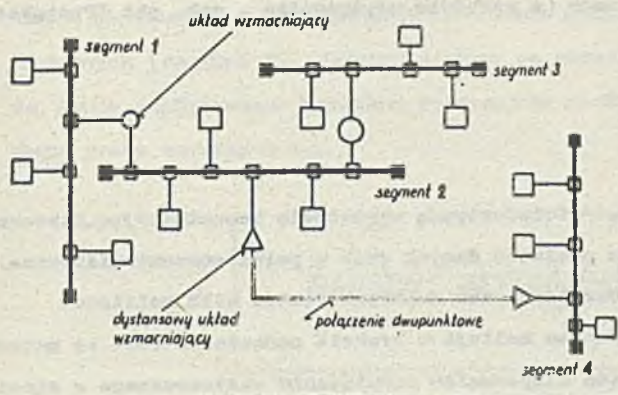
Przykład realizacji fizycznej dołączenia stacji do magistrali jest podany na rys. 2. Zacisk i blok N-O



Rys. 2. Przykład realizacji dołączenia stacji do magistrali

są umieszczone w jednej obudowie. Jednostka sterująca może być wykonana na jednym pakiecie, umieszczonym w obudowie stacji. Zakres sieci i liczbę dołączanych stacji można zwiększyć, łącząc segmenty między sobą za pomocą:

- układów wzmacniających,
- dystansowych układów wzmacniających (umożliwiających połączenie dwupunktowe na odległość do 1 km),
- układów wzmacniających z filtracją adresu (przesyłających do następnego segmentu tylko pakiety przeznaczone do stacji dołączonych do tego segmentu),
- brzołek (gateway) do dołączania segmentu do innych sieci.



Rys. 3. Przykład sieci Ethernet złożonej z 4 segmentów

Sposób przesyłania danych

W Ethernet nośnikiem informacji jest całkowicie pasywna magistrala, tzn. ekranowy kabel koncentryczny (używany w technice telewizyjnej), zapewniający transmisję sygnałów w paśmie podstawowym. Przyjęcie takiego rozwiązania zwiększa niezawodność sieci: jeżeli sieć jest jedno-segmentowa, wówczas awaria pewnej stacji nie wpływa na pracę całego segmentu; jeżeli sieć składa się z więcej niż jednego segmentu, wówczas jej własności niezawodnościowe są nieco gorsze, ponieważ układy wzmacniające są aktywnym elementem drogi transmisyjnej - awaria układu wzmacniającego powoduje częściowe unieruchomienie sieci.

W Ethernet przyjęte:

- transmisję szeregową, półduplexową, asynchroniczną sygnałów cyfrowych z kodowaniem fazy,
- szybkość transmisji 10 Mb/s,
- komutację pakietów,
- maksymalną odległość między stacjami 2,5 km,
- maksymalną liczbę stacji w całej sieci - 1024.

Dane są przesyłane magistralą w postaci pakietów o zmiennej wielkości (rys.4). Na początku pakietu są wysyłane bity synchronizujące, umożliwiające dostrojenie się wszystkich modułów N-O, dołączonych do magistrali. Pole "typ pakietu" umożliwia identyfikację formatu przesyłanych danych. Pole to jest wykorzystywane wówczas, gdy sieć Ethernet współpracuje z sieciami o innej architekturze np. DECnet firmy Digital, SNA firmy IBM, publiczne sieci z komutacją pakietów z protokołem X-25. (CCITT). Bity

Liczba bajtów

1	Bity synchronizujące
6	Adres przeznaczenia
6	Adres pochodzenia
2	Typ pakietu
46 ÷ 1500	Dane
4	Bity CRC

Rys. 4. Budowa pakietu danych

Wymienione układy wzmacniające nie są buforowane. Przykład sieci Ethernet złożonej z 4 segmentów jest pokazany na rys. 3.

Przy łączeniu segmentów obowiązuje zasada, że dwa segmenty sieci mogą być połączone za pomocą tylko jednego układu wzmacniającego wymienionych typów (dla uniknięcia interferencji danych, które, ze względu na różne czasy propagacji, mogłyby dochodzić do stacji różnymi drogami, a zatem w różnych chwilach czasowych).

cyklicznej kontroli nadmiarowej (cyklicznej redundancy check-CRC) są generowane przez wielomian stopnia 32. CRS jest obliczana sprzętowo.

W sieci Ethernet nie ma sprzętowego potwierdzenia dobrego odebrania pakietu przez stację przeznaczenia. Kontrola CRC w stacji przeznaczenia pakietu umożliwia wykrycie błędów związanych z kanałem transmisyjnym. Zwiększenie prawdopodobieństwa dobrego dostarczenia pakietu jest możliwe

za pomocą odpowiednich protokołów wysokich poziomów (w warstwie użytkownika - zob. pkt "Protokoły komunikacyjne").

Sposób dostępu stacji do magistrali

=====

W sieci Ethernet kabel koncentryczny stanowi wielodostępną magistralę komunikacyjną. Sterowanie dostępem stacji do magistrali dla przesłania pakietów danych jest w pełni zdecentralizowane.

Zastosowana została metoda dostępu CSMA/CD (carrier sense multiple access with collision detection) tzn. - próbkowanie fali nośnej z wykrywaniem kolizji w trakcie nadawania. Jest to metoda dostępu oparta na współzawodnictwie stacji, będąca ulepszeniem rozwiązania zastosowanego w sieci telekomunikacyjnej ALOHA z kanałami radiowymi.

Obowiązują następujące zasady dostępu do magistrali. Wszystkie bloki N-0 dołączone do magistrali prowadzą stały nasłuch, czy w kanale odbywa się przesyłanie danych - przez wykrywanie obecności fali nośnej (określenie przyjęte przez analogię do kanału radiowego). Jeżeli blok N-0 stacji, która chce przesłać pakiet wykrywa obecność fali nośnej - wówczas przesłanie pakietu jest zabronione i następuje oczekiwanie na wolną magistralę. Z chwilą, kiedy blok N-0 stwierdzi brak fali nośnej - dokonywane jest przesłanie pakietu. Równocześnie blok N-0 stacji wysyłającej pakiet porównuje sygnał przesyłany magistralą z sygnałem wysyłanym. Jeżeli te sygnały są zgodne - przesyłanie bitów pakietu jest kontynuowane, aż do przesłania pełnego pakietu.

Niezgodność obu sygnałów oznacza, że nastąpiła kolizja wysyłanego pakietu z pakietem (lub pakietami) innej stacji, która rozpoczęła przesyłanie pakietu dokładnie w tej samej chwili, co opisywana stacja. Wówczas zostaje przerwane przesyłanie bitów pakietu. Wobec przyjętej zasady wykrywania obecności fali nośnej, kolizja występuje najczęściej po zakończeniu przesłania jakiegoś pakietu przez magistralę, jeżeli w trakcie przesyłania tego pakietu co najmniej dwie stacje chciały przesyłać swoje pakiety.

Próba retransmisji pakietu jest podejmowana po upływie losowo określonego czasu opóźnienia retransmisji. Przy rozpoczynaniu próby przesyłania pakietu średni czas opóźnienia ewentualnej retransmisji jest równy maksymalnemu czasowi między rozpoczęciem transmisji i wykryciem kolizji. Gdy próba przesłania zakończy się kolizją, jednostka sterująca zwiększa dwukrotnie średni czas opóźnienia retransmisji (binary exponential backoff algorithm).

Opisana procedura dostępu może być stosowana w kanale o dowolnym nośniku informacji (kabel koncentryczny, para skręcona, światłowód, kanał radiowy).

Niezależnie od opisanej decentralizacji sterowania dostępem do sieci, zapewniony jest jednaki priorytet dla wszystkich stacji, tj. obowiązuje ograniczenie maksymalnej wielkości pakietów i jednaki algorytm doboru czasów opóźnienia retransmisji.

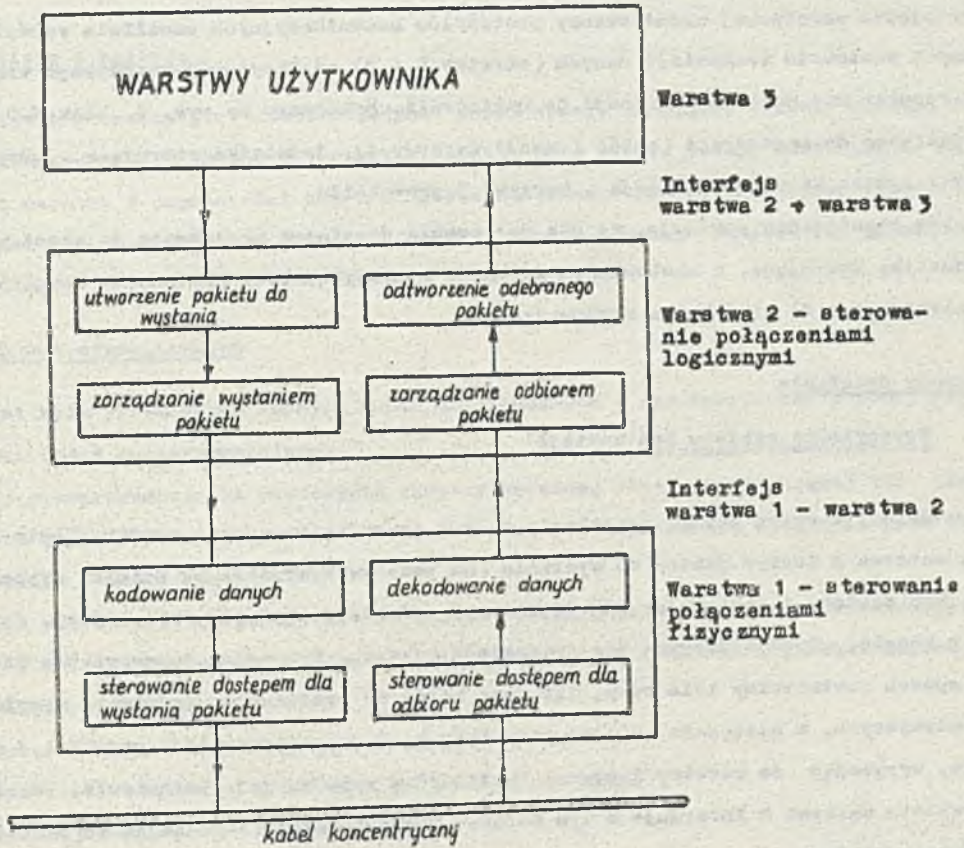
Protokoły komunikacyjne

=====

W sieciach telekomunikacyjnych stosowana jest norma ISO [4], określająca architekturę otwartych sieci komputerowych (Open Systems Interconnection). Przewiduje się 7-warstwową strukturę protokołów komunikacyjnych.

W sieci Ethernet przyjęto architekturę warstwową zgodną funkcjonalnie z normą ISO-OSI w zakresie

dwóch pierwszych warstw (rys.5): sterowania połączeń fizycznych (warstwa 1) i sterowania połączeń logicznych (warstwa 2). Warstwy wyższe są określone łącznie jako warstwa użytkownika (warstwa 3) i nie są ściśle zdefiniowane. Natomiast precyzyjnie zdefiniowano interfejsy międzywarstwowe oraz funkcje realizowane przez warstwy 1 i 2.



Rys. 5. Struktura warstwowa protokołów sieci Ethernet

Interfejs warstwa 2 + warstwa 3 obejmuje żądanie wysłania pakietu i dostarczenie słowa stanu, dotyczącego transmisji.

Warstwa 2, odpowiedzialna za sterowanie dostępem według zasad CSMA/CD, zapewnia realizację następujących funkcji

- utworzenia pakietu do wysłania (zamiana danych na postać szeregową, obliczenie CRC),
- zarządzania wysłaniem pakietu (nadzorowanie obecności fali nośnej, wysyłanie bitów zakończenia magistrali, obliczanie czasu opóźnienia retransmisji),
- odtworzenia bajtów danych z odebranego pakietu (kontrola poprawności pakietu: długości i CRC, kontrola zgodności adresów, zamiana danych szeregowych na bajty),
- zarządzania odbiorem pakietu (zbiórka kolejnych bitów).

Interfejs warstwa 1 + warstwa 2 obejmuje:

- przesyłanie sygnałów "kollzja", "obecność fali nośnej", "trwanie transmisji",
- wymianę ciągów bitów - do wysłania i po odbiorze,
- inicjowanie transmisji bitów.

Warstwa 1 zapewnia realizację następujących funkcji:

- kodowania i dekodowania danych,
- generowania bitów synchronizujących (preamble),
- sterowania dostępem do magistrali w celu wysyłania i odbierania bitów pakietu,
- wykrywania kolizji i obecności fali nośnej.

Przyjęcie warstwowej architektury protokołów komunikacyjnych umożliwia rozdzielenie funkcji logicznych protokołu transmisji danych (warstwy 2 i 3) od fizycznej realizacji transmisji danych.

W rozwiązaniu dołączenia stacji do magistrali, pokazanym na rys. 2, blok N-0 realizuje sterowanie dostępem do magistrali (część funkcji warstwy 1), jednostka sterująca - pozostałe funkcje warstwy 1 i całą warstwę 2, stacja - warstwę 3 protokołów.

Takie rozwiązanie powoduje, że dla dołączenia dowolnego urządzenia do sieci wystarczy opracować jednostkę sterującą, z zachowaniem warunków kompatybilności protokołów komunikacyjnych (co najmniej warstwy 1 i 2) oraz kabla sprzęgającego.

Zasady działania

Przesyłanie pakietu bez kolizji

Gdy warstwa użytkownika zażąda przesłania danych (z podaniem adresu przeznaczenia i typu pakietu), wówczas warstwa 2 tworzy pakiet do wysłania (na podstawie wyjściowej kolejki systemowej), dodając adresy, typ pakietu i obliczając CRC. Warstwa 2 nadzoruje obecność fali nośnej. Jeżeli nie ma transmisji w kanale, wówczas zaczyna się transmisja pakietu. Procedura "przesyłanie bitu" jest wywoływana w sposób powtarzalny tyle razy, ile jest bitów do przesłania. Warstwa 1 wysyła ciąg bitów synchronizujących, a następnie kolejno bity danych. Podczas transmisji warstwa 1 nadzoruje kolizje w kanale, wysyłając do warstwy 2 sygnał "kolizja" w wypadku jej wystąpienia. Po zakończeniu transmisji pakietu warstwa 2 informuje o tym warstwę 3 słowem stanu, zezwalając na ewentualne przesłanie następnego pakietu.

Odbiór pakietu bez kolizji

Warstwa 1 odbiera bity synchronizujące, umożliwiające dostrojenie odbiornika. Do warstwy 2 wysyłany jest sygnał "obecność fali nośnej". Od tej chwili przychodzące bity danych (procedura "odbior bitu" jest wywoływana przez warstwę 2) są dekodowane przez warstwę 1 na postać binarną i przekazywane do warstwy 2.

Po zaniku sygnału "obecność fali nośnej" warstwa 2 analizuje zgodność adresu przeznaczenia pakietu z adresem stacji, a także sprawdza poprawność pakietu - zgodność CRC i długość pakietu (powinna być całkowita liczba bajtów). Jeżeli jest zgodność adresów i pakiet jest poprawny (w opisanym wyżej sensie), wówczas odebrane bajty danych są przekazane do warstwy 3 wraz ze słowem stanu.

Przesyłanie pakietu z kolizją

Jeżeli po rozpoczęciu przesyłania pakietu warstwa 1 wykryje kolizję, wówczas przesyła do warstwy 2 sygnał "kolizja". Warstwa 2 przesyła przez pewien określony czas bity "zakłócenia magistrali" (jam), aby

wykrycie kolizji było możliwe przez wszystkie stacje, których pakiety brały udział w kolizji (collision consensus enforcement). Następnie warstwa 2 przerywa transmisję i przygotowuje retransmisję, która odbędzie się po czasie ustalonym przez warstwę 2. Po upływie tego czasu podejmowana jest próba retransmisji. Wskutek kolizji kanałem zostaje przesłany niekompletny, "obcięty" pakiet.

Odebranie pakietu z kolizji

Wykrycie kolizji w kanale jest możliwe jedynie przez stację nadającą. Dlatego warstwa 1 stacji odbierającej traktuje pakiet niekompletny wskutek kolizji jako ważny. Stacja odbierająca, na podstawie badania przez warstwę 2 poprawności CRC i długości odebranego pakietu, wykrywa odebranie niekompletnego pakietu (truncated packet filtering), a następnie oczekuje na powtórne przesłanie kompletnego pakietu.

Wyniki badań eksperymentalnych

Sieć Ethernet była przedmiotem różnych badań analitycznych i symulacyjnych. W pracy [2] przedstawione są wyniki badań eksperymentalnych.

Badania były przeprowadzane na prototypie eksperymentalnej sieci, zawierającej 120 stacji dołączonych do kabla koncentrycznego o długości 550 m. Pakiety o maksymalnej wielkości pola danych 512 B były przesyłane z szybkością 2,94 Mb/s.

W skład sieci wchodziły: komputery wykorzystywane przez pojedynczych użytkowników (głównie komputery Alto), 2 maszyny pracujące w podziale czasu, terminale, drukarki i pamięci masowe, a także kilka furtek międzysieciowych.

Sieć była wykorzystywana do następujących zadań:

- przesyłanie zbiorów danych do drukarek i do pamięci masowych,
- wykonywanie specjalnych programów wielomaszynowych,
- dostęp do dzielonych baz danych,
- zdalna diagnostyka elementów sieci,
- zdalne ładowanie programów,
- dostęp terminali do maszyn pracujących w podziale czasu.

W celu przeprowadzenia badań sieci opracowano odpowiednie programy testujące i nadzorujące pracę sieci. Wspólna magistrala bardzo ułatwia zbieranie informacji o pakietach przesyłanych magistralą.

Przeprowadzono trzy rodzaje eksperymentów z siecią.

Badania sieci w warunkach normalnego obciążenia

W ciągu doby magistralą przesyłane jest 2,2 mln pakietów - 300 mln B. Przesłanie do stacji odbiorczej błędnego pakietu zdarza się raz na 2 mln pakietów. Maksymalne obciążenie sieci mierzone w okresie 1 s wynosi 32,4 % pojemności kanału transmisyjnego. Rozkład długości pakietów przesyłanych magistralą może być aproksymowany rozkładem dwupunktowym: pakiety o długości 30 bajtów z prawdopodobieństwem 0,2 (głównie potwierdzenia odebrania pakietów i dane przesyłane przez terminale) oraz pakiety o długości 544 bajtów z prawdopodobieństwem 0,8 (głównie przesyłanie zbiorów danych).

Stwierdzono 1300 różnych par stacji wymieniających między sobą pakiety. 72 % pakietów jest przesyłanych w ramach sieci, 28 % pakietów jest przesyłanych furtkami międzysieciowymi.

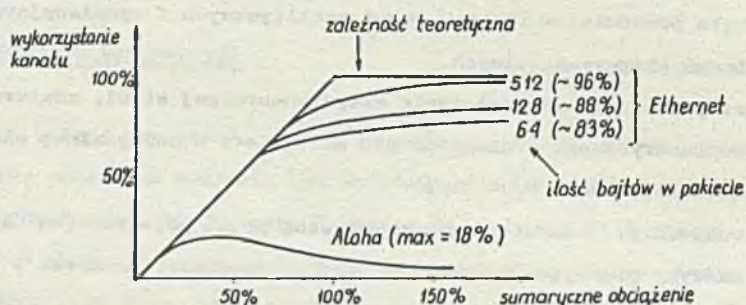
Średni czas między odbiorem kolejnych pakietów przez daną stację wynosi 59,5 ms.

Prawdopodobieństwo kolizji: 99,18 % pakietów jest przesyłanych bez kolizji, 0,79 % pakietów jest wysyłanych z opóźnieniem (wskutek zajętości magistrali), 0,03 % pakietów ulega kolizji.

69 % bitów przesyłanych przez magistralę to użytkowe bity danych, pozostałe 31 % to bity pomocnicze (nagłówki pakietów sieci Ethernet i międzysieciowe, CRC, pakiety potwierdzeń itp.).

Badanie sieci w warunkach dużego obciążenia

Dla celów eksperymentalnych, w sposób sztuczny, stworzono warunki, w których sumaryczne obciążenie magistrali przez stacje przysyłające pakiety może wynosić od 0 do 150 % pojemności kanału transmisyjnego.



Rys. 6. Wykorzystanie kanału transmisyjnego w warunkach dużego obciążenia

Na rys. 6 pokazana jest teoretyczna zależność wykorzystania kanału od sumarycznego obciążenia: wprost proporcjonalna do 100 % sumarycznego obciążenia oraz nasycenie - pełne wykorzystanie kanału dla większych sumarycznych obciążeń (naturalnie, nie wszystkie pakiety mogą być wówczas przesłane przez kanał). Badania wykazały, że jest to dobra aproksymacja rzeczywistej sytuacji (rys. 6); dla sumarycznych obciążeń powyżej 90 % pojemności kanału wykorzystanie kanału rośnie wraz z długością pakietów, osiągając wartość 96 % dla pakietów o długości 512 bajtów. Jest to bardzo dobry wynik; dla porównania: maksymalne wykorzystanie kanału w sieci ALOHA wynosi tylko 18 %. Sieć Ethernet jest stabilna w sensie podanym w punkcie "Zasady budowy sieci Ethernet" (poza gdy sieć ALOHA nie jest). Zbadano także równomierność dostępu stacji do magistrali. Przy sumarycznym 100 % obciążeniu (10 stacji, każda z obciążeniem 10 %) uzyskano 94 % wykorzystania kanału, przy czym wykorzystanie poszczególnych stacji zmieniało się od 9,3 % do 9,6 %.

Badanie sieci w warunkach skrajnego przeciążenia

Dla celów eksperymentalnych stworzono warunki, które nigdy w praktyce nie wystąpią: każda z 64 stacji dołączonych do sieci generowała w sposób ciągły pełne obciążenie kanału transmisyjnego.

Zgodnie z modelem analitycznym podanym w pracy [1] stwierdzono, że wraz ze wzrostem liczby stacji wykorzystanie kanału tylko nieznacznie maleje, np. dla pakietów o długości 512 bajtów

wykorzystanie kanału jest ciągle wyższe od 97 %. Stwierdzono stabilność sieci i równomierny dostęp stacji do kanału.

Uwagi końcowe

Ethernet jest obecnie "klasycznym" rozwiązaniem sieci lokalnej. Prowadzone są prace nad dalszym rozwojem sieci. Opracowano układy VLSI realizujące dwie pierwsze warstwy protokołów komunikacyjnych.

Na podstawie doświadczeń z badaniem i eksploatacją sieci Ethernet, począwszy od 1980 r. jest opracowywana norma na sieć lokalną - praca IEEE P 802, Local Network Standard. W pracy tej biorą udział, między innymi, firmy Xerox, Digital i Intel.

Literatura

- [1] Metcalfe R.M., Boggs D.R. - Ethernet: Distributed packet switching for local computer networks. Comm.ACM 1976 t.19 nr 7 s.395-404
- [2] Shoch J.F., Hupp J.A.: Measured performance of an Ethernet local network. Comm.ACM 1980 t.23 nr 12 s. 711-721
- [3] Monnier P.: Ethernet: réseau local à haute vitesse ouvert sur l'extérieur. Minis et micros 1981 nr 144, s. 31-36
- [4] ISO T097/SC16/N719 - Open Systems Interconnection Reference Model.

Biuletyn Informacyjny NAUKI I TECHNIKI KOMPUTEROWE

dr inż. Stanisława BONKOWICZ-SITTAUER
dr inż. Stanisław GRODZICKI

Instytut Maszyn Matematycznych

Oprogramowanie biblioteczne minikomputera MERA 400

Wstęp

Szerokie stosowanie komputerów, a zwłaszcza minikomputerów wiąże się z istnieniem takich narzędzi programistycznych dla danego sprzętu, które umożliwią użytkownikom łatwe opracowanie własnych wyspecjalizowanych programów [1] [2].

Istnienie, jakość i zakres takich narzędzi dla danego komputera może decydować o atrakcyjności konkurencyjnej sprzętu, co ma szczególne znaczenie w warunkach gospodarki wolnokonkurencyjnej. Natomiast w warunkach zmonopolizowanej gospodarki socjalistycznej obowiązkiem producenta kierującego się zasadami interesu społecznego jest zadbanie o dobre oprogramowanie swoich wyrobów. Nawiasem warto też zauważyć, że oprócz dbania o istnienie i stały rozwój oprogramowania danego komputera, firma produkująca sprzęt powinna też dbać o właściwe informowanie wszystkich aktualnych i potencjalnych użytkowników o stanie i zakresie całego istniejącego oprogramowania [3].

Obecnie dla minikomputera MERA 400 istnieje już sporo różnorodnego oprogramowania zarówno podstawowego jak i narzędziowego; istnieje też pewna ilość oprogramowania użytkowego [4]. Należy przy tym zaznaczyć, że Zakłady im. J. Krasiekiego uznają za standard system operacyjny SOM-3 i związane z nim pozostałe oprogramowanie poziomu podstawowego i narzędziowego (bibliotecznego). Są to:

- Podstawowa Biblioteka Matematyczna, której podprogramy są wykorzystywane przez FORTRAN jako funkcje standardowe, mogą być także dołączone do programów napisanych w ASSEMBLERZE. Biblioteka ta obejmuje takie zagadnienia, jak funkcje trygonometryczne liczb zmiennoprzecinkowych i zespolonych; pierwiastek kwadratowy liczby zmiennoprzecinkowej pojedynczej i podwójnej precyzji oraz liczby zespolonej; wartości bezwzględne; logarytmy, wydzielenie części całkowitej liczby; zamiana liczby całkowitej na zmiennoprzecinkową i na odwrót; reszta z dzielenia; największa i najmniejsza wartość z ciągu liczb zmiennoprzecinkowych i całkowitych i in.
- Biblioteka Pomocnicza Podprogramów - obejmuje podprogramy wykorzystywane przez programy tworzone za pomocą translatora FORTRAN IV MERA 400, jak również przez podprogramy Podstawowej Biblioteki Matematycznej. W języku FORTRAN nie ma możliwości odwołania się bezpośrednio do któregośkolwiek z podprogramów Biblioteki Pomocniczej, natomiast można odwoływać się z programu napisanego w makroassemblerze. Podprogramy tej biblioteki dotyczą m.in. takich zagadnień, jak: dodawanie i odejmowanie liczb zmiennoprzecinkowych podwójnej precyzji, liczb zespolonych, odczytanie liczb pojedynczej i podwójnej precyzji, negacja arytmetyczna liczb pojedynczej i podwójnej precyzji oraz zespolonych, iloczyn dwóch liczb podwójnej precyzji i zespolonych, standardowy program obsługi błędów i in.
- Fortranowska Biblioteka Automatyki (FBA) jest przeznaczona do realizacji zadań współpracy minikomputera MERA 400 z urządzeniami systemu PI.

- Fortranowska Biblioteka Czasu Rzeczywistego i Obsługi Zadań (FBCROZ) zawiera podprogramy, które umożliwiają pisanie i eksploatację programów czasu rzeczywistego oraz sterowanie zadaniami.
- Fortranowska Biblioteka Funkcji Binarnych (FBFB) jest przeznaczona do wykonywania działań logicznych na argumentach typu całkowitego, traktowane jako informacja binarna.
- Fortranowska Biblioteka Konwersji (FBK) zawiera podprogramy konwersji różnych typów, w szczególności możliwe są konwersje parametrów binarnych na kody ISO-7 i CAN i odwrotnie.
- Fortranowska Biblioteka Wejścia/Wyjścia (FBWW) zawiera podprogramy, umożliwiające organizowanie wejścia/wyjścia przez operacje: otwierania, zamykania i przydzielenia strumienia wejścia/wyjścia - czytania i pisanie rekordów fizycznych na określone strumienie wejścia/wyjścia; pozycjonowania nośników przydzielonych do strumienia wejścia/wyjścia.

Zakres forttranowskiej biblioteki matematyczno-statystycznej

Wspomniane we wstępie pojęcie oprogramowania narzędziowego może być rozumiane bardziej lub mniej szeroko; zawsze jednak należy tu uwzględnić różnorodne biblioteki (zob. Wstęp), przy czym większość firm komputerowych uznaje za absolutnie obowiązujące istnienie biblioteki matematyczno-statystycznej. Ponieważ początkowo w ramach oprogramowania związanego z systemem SOM-3 biblioteka taka nie istniała, należało lukę tę uzupełnić. W 1979 r. ówczesne CNPTKIP, a obecne Zakłady Wytwórcze Przyrządów Pomiarowych i Systemów Minikomputerowych Im. J. Krasickiego, powierzyły Pracowni Oprogramowania Zastosowań IMM wykonanie takiej biblioteki.

Po przeanalizowaniu analogicznych bibliotek komputerów IBM, ICL, ODRA-1300, ODRA-1204 i minikomputerów PDP, Wang, NOVA, K-202 oraz po przeanalizowaniu wyników wstępnego rozpoznania potrzeb wybranych, typowych użytkowników, przedstawił onowodawcy projekt takiej biblioteki [5]. Projekt ten został przyjęty i zrealizowany do końca 1982 r. Biblioteka w tej postaci jest przydatna dla licznych zastosowań, głównie naukowych i inżynierskich, jest więc propagowana [6] i rozpowszechniana zarówno przez ZWPPISM jak i bezpośrednio przez IMM^{m/}.

Przystępując do prac nad planowaną biblioteką założono, że zbiór podprogramów bibliotecznych jest otwarty, tzn. że początkowo zostanie wybrany powien podstawowy zestaw tematyczny zagadnień, który sukcesywnie może być uzupełniony przez opracowanie nowych podprogramów bądź zakup podprogramów, lub programów już wykonanych przez inne ośrodki komputerowe w kraju. Pozwoliło to na proste wykonanie zarówno wspomaganego rozszerzenia, jak i na wszelkie dalsze rozszerzenia, np. o kolejne działy matematyki czy statystyki lub o nowe grupy procedur organizacyjnych. Te ostatnie mogą z kolei rozwijać się w tzw. generatory, np. rysunków czy dialogów, w języki problemowe lub całe pakiety problemowe.

Drugą ważną zasadą, którą kierowano się przy projektowaniu zakresu merytorycznego omawianej biblioteki było dążenie do minimalizowania różnic między oprogramowaniem minikomputera MERA 400 a oprogramowaniem paru bardziej rozpowszechnionych w kraju minikomputerów, a przede wszystkim PDP, traktowanego jako wzorzec dla SM.

^{m/} zamówienia należy kierować na adres:

Starano się wreszcie tak dobrać poszczególne moduły biblioteczne aby służyły do rozwiązywania zagadnień najczęściej stosowanych w nauce i praktyce technicznej, uwzględniając przy tym obecny najnowszy stan wiedzy w danych dziedzinach matematyki i statystyki. Dlatego też wiele algorytmów stosowanych np. w bibliotece PDP zostało zastąpionych nowszymi, zaczerpniętymi z odpowiednich publikacji [8], [9].

Biorąc pod uwagę powyższe przesłanki zaproponowano następujący zakres tematyczny:

- algebra liniowa,
- wielomiany i szeregi,
- równania różniczkowe
- całkowanie numeryczne,
- obliczanie funkcji, aproksymacja, interpolacja
- elementy geometrii analitycznej na płaszczyźnie,
- obliczenia liczbowe (pomocnicze),
- statystyka (analiza danych, nieparametryczne testy istotności, generacja liczb losowych),

Podprogramy biblioteki matematyczno-statystycznej mogą być stosowane w pracach naukowych do opracowania wyników pomiarów eksperymentalnych i laboratoryjnych, do opracowania wyników badań w medycynie, geologii, geodezji, rolnictwie itp. Podobnie podprogramy te mają szerokie zastosowanie do obliczeń realizowanych w projektowaniu.

Spośród metod matematycznych, które są oprogramowane w bibliotece minikomputera MERA 400 warto zwrócić uwagę na:

- operacje macierzowe, a więc np. różne metody rozwiązywania równań liniowych (układ ok. 70 równań) - metoda Chlesky'ego, metoda Gaussa, metoda najmniejszych kwadratów, metoda przybliżona; odwracanie macierzy, określanie wartości własnych i wektorów własnych macierzy (są to metody Jacobiego i transformacji), metoda QL i in.;
 - działania na wielomianach, np. obliczanie wartości wielomianu i jego pochodnej, całkowanie wielomianu, obliczanie wielomianów Legendre'a, arytmetyka wielomianów, rozwiązywanie równań algebraicznych 2 i 3 stopnia i in.
 - obliczanie całek, np. metoda Simpsona, kwadratura Romberga, kwadratura Gaussa - Legendre'a, metoda Newtona-Cotesa, metoda Monte Carlo i in.
 - rozwiązywanie równań różniczkowych - metodą Runge-Kutty czwartego rzędu, metoda Mersona, metoda Hamminga i in.
 - elementy geometrii analitycznej na płaszczyźnie - równanie prostej, odległości punktu od prostej i między prostymi równoległymi, równanie dwusiecznej kąta, kąt ze współrzędną, transformacja współrzędnych i in.
 - pomocnicze obliczenia liczbowe - obliczanie silni i jej odwrotności, obliczanie liczby kombinacji z liczby naturalnej N , obliczanie pola powierzchni dowolnej figury płaskiej, zamiana współrzędnych prostokątnych na biegunowe i odwrotnie, obliczanie azymutu ze współrzędnych i in.
 - funkcje specjalne - funkcje Bessela pierwszego i drugiego rodzaju oraz zmodyfikowane funkcje gamma, całkę eliptyczną pierwszego i drugiego rodzaju, całkę wykładniczą, całki sinusa i cosinusa całkowitego, całkę Fresnela;
 - optymalizację w zakresie algorytmów optymalizacyjnych - algorytm błędzący, algorytm Coopera - Glasa, algorytm losowo-gradientowy, algorytm Resenbrooka i algorytm Wilde'a;
 - równania różniczkowe - te same metody co uprzednio, ale opracowano podprogramy w podwójnej pozycji.
- Natomiast w zakresie metod statystycznych należy wymienić zadania dotyczące analizy danych, obliczanie wartości średnich, odchylenia standardowego, minimalnej i maksymalnej wartości zmiennej, obliczanie

współczynników korelacji, obliczanie regresji wielorakiej, obliczanie wariancji i kowariancji, obliczanie statystyki - kwadrat Pearsona, testy: Kandała, Spearmana, Q - Colbrana, generatory liczb losowych - o rozkładzie normalnym i równomiernym oraz

- obliczanie liczebności przedziałów klasowych dla jednej zmiennej i dla klasyfikacji dwukierunkowej, przy danej macierzy obserwacji,
- obliczanie średniej arytmetycznej zmiennych w każdej grupie oraz zsumowania macierzy dyspersyjnej dla wszystkich grup analizy dyskryminacyjnej,
- wykonanie ortogonalnych rotacji macierzy czynników,
- znajdowanie szeregu chronologicznego,
- obliczanie wartości średnich, odchyłeń standardowych, współczynników kurtozy i skośności, współczynników regresji i korelacji, błędów standardowych regresu i in.

Jak widać z tego skrótego przeglądu zagadnień - zestaw modułów bibliotecznych ułatwi rozwiązanie wielu problemów naukowych i praktycznych.

- W dziale geometrii analitycznej warto też wymienić takie zagadnienia, jak obliczanie kąta między prostymi, obliczanie współczynników równań prostych do danej prostej prostopadłej i równoległej, obliczanie współrzędnych punktu przecięcia prostej z okręgiem, współrzędne przecięcia dwóch okręgów, transformacja współrzędnych zadanych punktów z jednego układu na drugi i in.

Ponadto wprowadzona została grupa procedur organizacyjnych o łącznej nazwie JESTA [11]. Pozwala ona na proste korzystanie z podprogramów statystycznych użytkownikowi nie znającemu programowania w języku FORTRAN.

Ogólne zasady korzystania z biblioteki

Fortranowska biblioteka matematyczno - statystyczna jest zapisana na dysku wymiennym jako biblioteka użytkownika. Sposób korzystania z takiej biblioteki jest podany w dokumentacji systemu MERA 400 [12].

Pszczególnie moduły biblioteczne są opracowane w postaci podprogramów typu SUBROUTINE lub FUNCTION i są wołane zgodnie z zasadami języka programowania FORTRAN.

Każdy z podprogramów jest odpowiednio i jednolicie udokumentowany [3]. Dokumentacja modułów bibliotecznych jest opracowywana w dwóch poziomach: poziom pierwszy zawiera dokumentację użytkową podprogramów, która ma wszelkie informacje potrzebne dla użytkownika, poziom drugi obejmuje dokumentację szczegółową podprogramów, w której są zawarte informacje niezbędne przy konserwacji programu (poprawianie błędów, rozszerzenie itp.).

Całość dokumentacji użytkowej jest wykonana w 7 zeszytach. Praktycznie w jednym zeszycie zawarty jest opis użytkowy podprogramów z kilku dziedzin matematyki.

Przedmiotem dystrybucji jest oprócz nośnika dystrybucyjnego dokumentacja użytkowa podprogramów, z której użytkownik uzyska niezbędne informacje potrzebne do zastosowania modułu bibliotecznego.

Uwagi końcowe

Jak już wspomniano, użytkownik stosujący komputer do rozwiązywania swoich merytorycznych zagadnień zainteresowany jest w istnieniu takiego oprogramowania narzędziowego, które pozwoli mu tworzyć

własne bardzo wyspecjalizowane programy w sposób prosty, a co ważniejsze nie wymagający od niego żadnego lub prawie żadnego przygotowania w obsłudze i programowaniu. Z drugiej strony jednak zawsze będzie istniała spora grupa użytkowników (pracownicy naukowcy, projektanci itp.), którzy nie będą obojętni na gotowe, stosunkowo sztywne programy, pakiety czy systemy problemowo zorientowane. Zakres i różnorodność ich zainteresowań i potrzeb będzie wymagała powstawania coraz to nowych programów szczegółowych. Z myślą o takim użytkowniku powstają na świecie różnorodne narzędzia programistyczne, generatory, języki problemowe itp. I w takim właśnie oprogramowaniu narzędziowym powinna być z czasem wyposażona i MERA 400. Powinny powstać zarówno generatory dialogów, rysunków i wydruków, jak i języki programowo ukierunkowane na poszczególne działy nauki, techniki lub zarządzania.

Literatura

- [1] Bonkowiak-Sittauer W.: Problemy Oprogramowania Projektującego. Biul. Inf. BIOSK 1981 nr 2
- [2] Bonkowiak-Sittauer W.: Narzędzia programistyczne dla komputerowo wspomaganego projektowania. Ref. na konf. pt. "Zastosowanie komputerów w przemyśle" - Szczytno 1981 (maszynopis)
- [3] Bonkowiak-Sittauer W., Pokulniewicz M.: - Opracowanie zasad prowadzenia i rozpowszechniania informacji o oprogramowaniu minikomputera MERA 400 PIRIO/MERA400.IMM, kwiecień 1981. Sprawozdanie z pracy planowej.
- [4] Gutowska H.: Próba analizy przydatności firmowego oprogramowania minikomputera MERA400 do wspomagania prac inżynierskich. Ref. na konf. pt. "Automatyzacja prac w projektowaniu. Poznań 1981.
- [5] Projekt biblioteki, etap I .R/64, IMM, wrzesień 1979 (maszynopis)
- [6] Biblioteka matematyczno-statystyczna podprogramów fortranowskich MERA400, SOM3, IMM Warszawa 1981 prospekt
- [7] Dokumentacja Użytkowa Podprogramów Matematycznych. IMM 1981 (maszynopis)
- [8] Wilkinson J.H., Reinsch C.: Linear Algebra. Springer Verlag 1971
- [9] Hartley H.D.: Mathematical methods for digital computers. John Wiley and Sons 1962
- [10] Projekt rezerwy fortranowskiej biblioteki matematyczno-statystycznej. IMM czerwiec 1981 (maszynopis)
- [11] Pawlik R.: Minikomputerowy system wspomagania prac inżynierskich (projektowych) - projekt realizacji. IMM 1981 (maszynopis)
- [12] Dokumentacja techniczno-ruchowa system MERA400 (maszynopis)

Biuletyn Informacyjny NAUKI I TECHNIKI KOMPUTEROWE

dr inż. Ryszard PAWLIK
Instytut Maszyn Matematycznych

Język do przetwarzania danych statystycznych

w bibliotece fortranowskiej systemu MERA 400

Jednym z celów każdego oprogramowania jest ułatwienie użytkownikowi korzystania z komputera. Niniejsze opracowanie jest jedną z prób takiego ułatwienia.

Korzystanie z oprogramowania opracowanego w formie tradycyjnej biblioteki podprogramów (np. w języku FORTRAN - typu SUBROUTIN) może pewnym użytkownikom stwarzać trudności. Opracowano więc narzędzie programowe język programowania, które trudności te znacznie zmniejsza. Język ten obsługuje opracowaną w Instytucie Maszyn Matematycznych fortranowską bibliotekę procedur statystycznych [1]. Biblioteka ta włączona została do systemu MERA 400. Przeznaczona jest dla użytkowników znających język programowania FORTRAN. Bibliotekę tworzy zestaw 25 procedur fortranowskich typu SUBROUTINE. Użytkownik chcąc skorzystać z biblioteki musi wykonać następujące czynności:

- wybrać ze zbioru procedur interesujące go procedury
- dokładnie poznać listę parametrów formalnych wybranych procedur
- napisać fragment programu zapewniający wprowadzenie danych niezbędnych dla rozwiązania problemu
- napisać fragment programu wywołujący poszczególne procedury biblioteki, dbając o ich poprawne wywołanie oraz wymianę informacji między nimi
- napisać fragment programu wyprowadzający uzyskane wyniki
- doprowadzić do poprawnej translacji napisanego programu
- usunąć w programie błędy, które ujawniają się w czasie wykonywania programu
- wykonać program dla uzyskania żądanych wyników

Wymienione czynności zajmują użytkownikowi bardzo dużo czasu. Czas ten traci on na czynności związane z programowaniem, a więc te, które nie są lub nie powinny być dla niego zasadnicze. Należy więc dążyć do skrócenia tego czasu.

Jedną z metod przyspieszających korzystanie z biblioteki procedur może być język ukierunkowany problemowo. Dysponując językiem ukierunkowanym problemowo użytkownik nie musi znać języka programowania FORTRAN, aby móc skorzystać z opracowanej biblioteki. Cechą charakterystyczną języka ukierunkowanego problemowo jest to, że pomimo jego sformalizowania jest on zbliżony do języka z dziedziny, którą opisuje. Użytkownik używając tego języka tworzy program rozwiązujący ten problem.

Język ukierunkowany problemowo JESTA

Każdy rozkaz języka JESTA zawiera słowo lub zdanie, które wyraża żądanie użytkownika wykonania pewnych obliczeń i jednocześnie identyfikuje ten rozkaz. Ogólna postać rozkazu może zawierać liczne warianty i rozgałęzienia, umożliwiające użytkownikowi pewną elastyczność postępowania. Rozkaz języka JESTA ma następującą ogólną budowę:

rozkaz : = nazwa rozkazu, słowo problemowe 1, ..., słowo problemowe N, lista ;

Uwaga: w każdym rozkazie musi wystąpić nazwa rozkazu, pozostałe składniki w niektórych rozkazach nie występują lub występują nie wszystkie.

Nazwa rozkazu - identyfikator określający, jakie czynności mają być wykonane. Czynności te realizowane są przez poszczególne podprogramy, przy czym z każdym rozkazem może być sprzężona jedna lub kilka procedur należących do biblioteki statystycznej. Jednocześnie różne rozkazy mogą powodować wykonanie tych samych procedur.

Słowo problemowe - identyfikator służący do rozgałęzienia rozkazu.

Lista - ciąg składający się z elementów rozdzielonych przecinkami. Elementami listy mogą być liczby całkowite, liczby rzeczywiste lub ciągi znaków rozpoczynających się od litery i nie dłuższych niż 6 znaków (ciągi te zwane będą dalej nazwami). Znak jest to element z alfabetu języka, który jest identyczny z alfabetem języka FORTRAN na maszynie MERA 400.

Lista rozkazów języka JESTA

SYSTEM*
CZYTAJ
WYPROWADZAJ
WSPOLCZYNNIK
POCZATKOWEOBLICZENIA
STATYSTYKA
KOWARIANCJA
WARIANCJA
PRAWDOPODOBIENSTWO
NORMOWANIE
TESTOWANIEHIPOTEZY
KONIEC

Program użytkownika zaczyna się rozkazem SYSTEM, a kończy rozkazem KONIEC. Między tymi rozkazami mogą znajdować się pozostałe rozkazy języka w dowolnej kolejności.

Poniżej została dokładniej opisana budowa poszczególnych rozkazów języka JESTA.

Przy opisie rozkazów zastosowano następujące reguły:

- podkreślenia liter w słowach problemowych mają takie samo znaczenie jak przy nazwach rozkazów - oznaczają minimalne skróty tych słów
- brak podkreśleń w słowie problemowym oznacza, że słowo to można opuścić, a skutek będzie taki jakby to słowo było użyte,
- każdy rozpoczęty nawias klamrowy oznacza, że dokładnie jeden element z tego nawiasu musi wystąpić

Rozkaz SYSTEM

Format ogólny

SYSTEM , STATYSTYCZNY ;

Rozkaz ten służy do wywołania systemu statystycznego i jest pierwszym rozkazem programu użytkownika. Rozkaz ten zawsze musi wystąpić dokładnie jeden raz i dokładnie w takiej postaci.

* Nazwy rozkazów można skracać. Litery podkreślone określają minimalny skrót nazwy rozkazu nie powodujący jego błędnej interpretacji.

Rozkaz CZYTAJ

Format ogólny:

<u>CZYTAJ</u> ,	{	<u>DYSK</u> ,lista;
		<u>MONITOR</u> ,
	{	<u>CALKOWITE</u> ,lista;
		<u>RZECZYWISTE</u> ,lista;
		<u>TEKST</u> ,lista;
	{	<u>TASMA</u> ,lista;
{		<u>DANESTALE</u> ,
		<u>LICZBA</u> ,lista;
		<u>TABLICA</u> ,lista;
{	<u>TEKST</u> ,lista	
	<u>PROGRAM</u> ,lista;	

Rozkaz CZYTAJ służy do wprowadzania do programu danych. W rozkazie tym uwzględniono różne typy urządzeń, z których mogą być wprowadzane dane oraz różne sposoby ich wprowadzania. Rodzaj i forma danych zostały dobrane do typu urządzenia i sposobu ich wprowadzania. Poszczególne słowa problemowe odpowiednio oznaczają:

DYSK - dane będą odczytywane z dysku. Na liście występującej po słowie DYSK muszą być podane informacje o tym, jakie dane odczytujemy i jaka jest ich struktura.

MONITOR - dane będą wprowadzane bezpośrednio z monitora lub konsoli operatorskiej. Poszczególne słowa problemowe kolejno oznaczają:

MONITOR,CALKOWITE - że wprowadzony będzie ciąg liczb całkowitych. Wielkość tego ciągu określa lista, która składa się z nazw zmiennych. Zmienne te zostaną uznane za całkowite.

Przykład 1. Rozkaz CZ, MO, CA, I, B, X; powoduje, że w czasie wykonywania programu na ekranie monitora lub konsoli operatorskiej pojawi się napis:

PODAJ WARTOSCI CALKOWITE ZMIENNYCH -I,B,X po którym należy wprowadzić trzy liczby całkowite; ich wartości będą przypisane kolejno zmiennym I,B oraz X.

MONITOR,RZECZYWISTE - że wprowadzony zostanie ciąg liczb rzeczywistych. Wielkość tego ciągu określa lista, która składa się z nazw zmiennych. Zmienne te będą uznane za zmienne rzeczywiste.

Uwaga 1

Słowo problemowe RZECZYWISTE można pominąć, ponieważ przy braku słowa problemowego po słowie MONITOR uważa się, że lista zawiera zmienne typu rzeczywistego.

Przykład 2. Rozkaz CZ, MO, W, X, D; powoduje, że w czasie wykonywania programu na ekranie monitora lub konsoli operatorskiej pojawi się napis:

PODAJ WARTOSCI RZECZYWISTE ZMIENNYCH - W,X,D po którym należy wprowadzić trzy liczby rzeczywiste np. 3.5, 2.7, 0.5. Wartości tych liczb zostaną przypisane kolejno zmiennym W,X,D, tzn. w tym wypadku W=3.5, X=2.7 a D=0.8.

MONITOR,TEKST - że wprowadzone będą dane alfanumeryczne. Lista zawiera ciąg zmiennych, na których dane te będą przechowywane. Na jednej zmiennej można przechować co najwyżej 6 znaków.

Przykład 3. Rozkaz CZ, MO, KO, X1, X2 powoduje, że w czasie wykonywania programu pojawi się napis:

PODAJ TEKST DO 12 ZNAKOW - po którym należy wprowadzić tekst składający się co najwyżej z 12 znaków razem ze wszystkimi zawartymi w nim spacjami.

Uwaga 2

Jeżeli tekst zawiera więcej znaków niż liczba zmiennych pomnożona przez 6, to nadmiarowe znaki będą pominięte. Jeżeli zaś tych znaków będzie mniej, to wprowadzony tekst uzupełniony zostanie spacjami.

TASMA - że dane będą wczytywane z taśmy papierowej. Lista określa strukturę danych.

DANESTALE - że dane zostaną wprowadzone do programu w trakcie pisania programu. Pozwala to na wprowadzenie pewnych udogodnień dla użytkownika. Poszczególne słowa problemowe kolejno oznaczają:

DANESTALE,LICZBA - że wprowadzony będzie ciąg liczb. Wielkość tego ciągu i jego strukturę określa lista. Lista składa się z par {nazwa , liczba}.

Uwaga 3

Nie deklaruje się tu typu liczby. Jeżeli wprowadzona liczba zawiera kropkę dziesiętną, to uważana jest za liczbę rzeczywistą, jeżeli kropka \cdot nie występuje, to jest to liczba całkowita.

Przykład 4. Rozkaz CZ,DA,A,5,X,7,ABC,8.5; powoduje, że w programie zmienna A przyjmuje wartość 5, zmienna X przyjmie wartość 7 a zmienna ABC wartość 8.5. Oczywiście zmienne A, X będą zmiennymi typu całkowitego a zmienna ABC będzie miała typ rzeczywisty. Słowo problemowe LICZBA, zgodnie z formatem ogólnym rozkazu CZYTAJ zostało pominięte.

DANESTALE,TABLICA - oznacza, że wczytywane liczby będą tworzyć tablicę. Lista po słowie problemowym TABLICA musi zawierać kolejno następujące elementy: nazwę, która będzie nazwą tablicy, bezpośrednio po nazwie deklarację rozmiarów tablicy, wartości poszczególnych elementów tablicy zapisane kolumnami. Rozmiar tablicy deklaruje się liczbami całkowitymi, oddzielonymi przecinkami i ujętymi w nawiasy np. A/3,2/ - oznacza 6-elementową tablicę o nazwie A, trzech wierszach i 2 kolumnach.

Uwaga 4

Przy deklaracji tablicy ciąg znaków występujący na liście może wyjątkowo zawierać więcej niż 6 znaków, przy czym sama nazwa nie może być dłuższa niż 6 znaków.

Przykład 5. Rozkaz CZ,DA,TA,ABCD/2,2/,5,6,7,8; zawiera napis ABCD/2,2/ o długości 9 znaków i jest poprawny, ponieważ nazwa ABCD zawiera tylko 4 znaki a pozostałe 5 znaków deklaruje rozmiar tablicy.

Uwaga 5

Podobnie jak przy słowie problemowym LICZBA nie deklaruje się typu tablicy. Typ tablicy określa typ wprowadzonych liczb. Jeżeli wśród wprowadzanych liczb wystąpi chociaż jedna liczba z kropką dziesiętną, to tablica otrzyma typ rzeczywisty, w przeciwnym razie tablica będzie typu całkowitego.

Uwaga 6

Tablice mogą być co najwyżej dwuwymiarowe.

Przykład 6. Rozkaz CZ,DA,TA,A/3,2/,5,6,2,3,8,6; powoduje, że w programie utworzona zostanie tablica A typu całkowitego, której elementy będą mieć następujące wartości:

$$A/1,1/=5 \quad , \quad A/1,2/=3$$

$$A/2,1/=6 \quad , \quad A/2,2/=8$$

$$A/3,1/=2 \quad , \quad A/3,2/=6$$

Uwaga 7

Jeżeli kolejne elementy tablicy są identyczne, to można taki element poprzedzić znakiem "x" a przed tym znakiem podać liczbę powtórzeń.

Przykład 7. Rozkaz CZ,DA,TA,B/3,3/,1,7.2,4*2,5.3,6,0.5; powoduje, że w programie utworzona zostanie tablica typu rzeczywistego, której elementy będą mieć następujące wartości:

B/1,1/=1 , B/1.2/=2 , B/1,3/=5.3,

B/2,1/=7.2, B/2,2/=2, B/2,3/=6,

B/3,1/=2, B/3,2/=2, B/3,3/=0.5

Uwaga 8

Jeżeli użytkownik poda za mało liczb, to tablica zostanie uzupełniona zerami, a jeżeli za dużo, to nadmiarowe liczby będą pominięte.

Przykład 8. Rozkaz CZ,DA,TA,X/5/,2,3; powoduje, że w programie utworzona zostanie tablica typu całkowitego, której elementy będą mieć następujące wartości:

X/1/=2,

X/2/=3,

X/3/=X/4/=X/5/=0

Przykład 9. Rozkaz CZ,DA,TA,M/4/,5.2,6,7,4*5.6; powoduje, że w programie utworzona zostanie tablica typu rzeczywistego, której elementy będą mieć następujące wartości:

M/1/=5.2, M/2/=6, M/3/=7, M/4/=5.6

DANESTALE,TEKST - oznacza, że będą wozytywane dane alfanumeryczne. Dane alfanumeryczne wozytywane są na zmienne proste typu rzeczywistego (nie potrzeba definiować typu tylko nazwę) lub tablice. Na jedną zmienną prostą lub jeden element tablicy można zapisać co najwyżej 6 znaków. Lista zawiera pary {nazwa, 'tekst'}.

Przykład 10. Rozkaz CZ,DA,TE,A, 'DANE', B, 'WYNIKI'; powoduje, że w programie zmienna A otrzyma wartość DANE, a zmienna B otrzyma wartość WYNIKI.

Przykład 11. Rozkaz CZ,DA,TE,A/2/, 'OBLICZENIA'; powoduje, że w programie utworzona zostanie tablica A, której odpowiednie elementy przyjmą wartości: A/1/=OBLICZ, A/2/=ENIA.

Uwaga 9

Gdy liczba znaków w komentarzu jest za duża, to nadmiarowe znaki są pomijane, a gdy za mała, to na końcu dopisywane są znaki spacji.

Przykład 12. Rozkaz CZ,DA,TE,I,JEZYKJESTA,B/3/,WARSZAWA; powoduje, że zmienna rzeczywista I przyjmie wartość JEZYKJ, a elementy tablicy B będą miały następujące wartości:

B/1/=WARSZA , B/2/=WA____, B/3/=_____

gdzie znak "_" oznacza spację (odstęp).

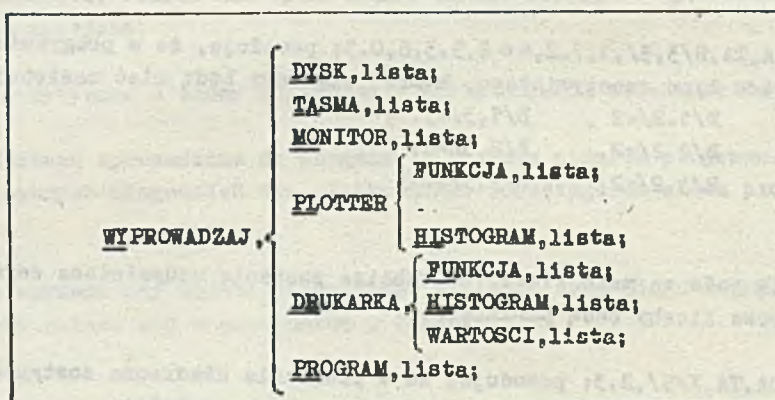
PROGRAM - oznacza, że do programu dołączony będzie podprogram napisany w języku FORTRAN (typu SUBROUTINE), wykonanie którego zapewni wprowadzenie do programu danych w postaci wymaganej przez daną sytuację obliczeniową. Lista po słowie problemowym program zawiera na pierwszym miejscu nazwę podprogramu, a następnie listę parametrów formalnych podprogramu.

Uwaga 10

Słowo problemowe PROGRAM przeznaczone jest dla użytkowników znających język programowania FORTRAN. Procedura fortranowska musi być uruchomiona i przetestowana wcześniej oraz, co jest szczególnie ważne, dołączona do biblioteki procedur fortranowskich, z którą współpracuje język JESTA.

Rozkaz WYPROWADZAJ

Format ogólny



Rozkaz WYPROWADZAJ służy do wyprowadzania wyników obliczeń. Podobnie jak w rozkazie CZYTAJ uwzględniono różne typy urządzeń wyjściowych i różną postać graficzną wyników.

Poszczególne słowa problemowe odpowiednio oznaczają:

DYSK - że wyniki zapisane zostaną na zbiorze dyskowym. Na liście występującej po słowie dysk - muszą być podane informacje, jakie dane zapisujemy na dysk oraz jaka jest ich struktura.

TASMA - że wyniki zapisane będą na taśmie papierowej. Lista określa strukturę tych wyników.

MONITOR - że wyniki wyprowadzone będą na monitor ekranowy. Lista zawiera ciąg nazw, których wartości mają być wyprowadzone. Na ekranie monitora wyświetlone zostaną nazwy żądanych zmiennych oraz ich wartości, jeżeli są to wartości liczbowe, w kolejności podanej na liście. Natomiast nazwy zmiennych, w których przechowywane są wartości alfanumeryczne nie będą wyświetlane.

Uwaga 11

Każdy nowy rozkaz WYPROWADZAJ powoduje, że wartości będą wyświetlane od nowego wiersza, a potem aż do wyczerpania listy ze zmianą wiersza, gdy zostanie zapełniony wiersz poprzedni.

Przykład 13. Zmienna A ma wartość 5, B=WYNIK. Rozkaz WY,MO,B,A; powoduje wyświetlenie na monitorze następującego napisu:

WYNIK: A=5

Przykład 14. Zmienne A i B mają takie same wartości jak w przykładzie 13. Piszemy dwa poniższe rozkazy WY,MO,B; WY,MO,A; powoduje to wyświetlenie następującego tekstu:

WYNIK;

A=5

Uwaga 12

Po wyświetleniu każdej wartości zmiennej robiony jest jeden odstęp.

Wyświetlane są tylko znaczące cyfry liczby przed kropką dziesiętną poprzedzone znakiem "=" oraz 5 pierwszych cyfr znaczących po przecinku.

PLOTTER - że wyniki mają być przedstawione w postaci graficznej. Słowo problemowe PLOTTER ma dwa

dalsze słowa problemowe, które odpowiednio oznaczają:

PLOTTER,FUNKCJA - że wynik będzie przedstawiony w postaci funkcji jednej zmiennej. Na liście występującej po słowie problemowym FUNKCJA podane muszą być informacje o tym, na jakiej zmiennej przechowywane są wartości funkcji i odpowiadające im argumenty, żądanej skali wykreślenia funkcji itp. dane niezbędne do czytelnego przedstawienia wykresu funkcji.

PLOTTER,HISTOGRAM - że wynik przedstawiony będzie w postaci histogramu. Na liście występującej po słowie problemowym HISTOGRAM muszą być podane informacje niezbędne do poprawnego i czytelnego przedstawienia histogramu.

DRUKARKA - że wyniki będą wyprowadzane na drukarkę. Po słowie problemowym DRUKARKA mogą wystąpić trzy dalsze słowa problemowe, które odpowiednio oznaczają:

DRUKARKA,FUNKCJA - że wynik będzie przedstawiony w postaci wykresu funkcji jednej zmiennej. Na liście występującej po tym słowie muszą być podane wszystkie informacje niezbędne do narysowania funkcji.

DRUKARKA,HISTOGRAM - że wynik będzie przedstawiony w postaci histogramu. Na liście występującej po tym słowie problemowym muszą być podane wszystkie informacje niezbędne do narysowania histogramu.

Uwaga 13

Wykres funkcji i histogram są przedstawiane z dokładnością możliwą do przedstawienia na drukarce. Budowa list po słowach problemowych PLOTTER,FUNKCJA będą identyczne. Podobnie budowa list po słowach PLOTTER,HISTOGRAM i DRUKARKA,HISTOGRAM także będą identyczne. Pozwoli to wyprowadzać rysunki na drukarkę lub plotter bez potrzeby zmian w programie.

DRUKARKA,WARTOSCI - że wyniki w postaci alfanumerycznej będą wyprowadzane na drukarkę. Na liście podanej po słowie problemowym WARTOSCI podaje się jakie zmienne i z jakimi komentarzami mają być wyprowadzone.

Uwaga 14

Obowiązują tu takie same zasady jak przy wyprowadzaniu wyników na monitor rozkazem: WYPROWADZAJ,MONITOR,lista;. Zasady te zostały zawarte w opisie słowa problemowego MONITOR rozkazu WYPROWADZAJ razem z uwagami 11 i 12 oraz zilustrowane podanymi tam przykładami 13 i 14.

PROGRAM - oznacza, że do programu będzie dołączony podprogram napisany w języku FORTRAN typu SUBROUTINE, wykonanie którego zapewni wyprowadzenie wyników w postaci żądanej przez użytkownika.

Uwaga 15

Słowo problemowe PROGRAM spełnia podobną funkcję jak słowo problemowe PROGRAM w rozkazie CZYTAJ. Obowiązują tu te same zasady, które są podane w opisie rozkazu CZYTAJ razem z podaną tam uwagą 10.

Rozkaz WSPÓLCZYNNIK

Format ogólny:

<u>WSPOLCZYNNIK,</u>	<u>KORELACJI,</u>	<u>SPEARMANA</u> , lista;
		<u>KANONICZNEJ</u> , lista;
		<u>ZWYKLEJ</u> , lista;
	<u>ZGODNOSCI</u> , lista;	
	<u>REGRESJI</u> , lista;	

Poszczególne słowa problemowe odpowiednio oznaczają:

KORELACJI, SPEARMANA - obliczanie współczynnika korelacji rank Spearmana między dwiema zmiennymi,

KORELACJI, KANONICZNEJ - przeprowadzenie analizy korelacji kanonicznej między dwoma grupami zmiennych.

KORELACJI, ZWYKLEJ - obliczenie wszystkich współczynników korelacji między grupami obserwacji wielu zmiennych.

ZGODNOSCI - obliczenie współczynnika zgodności Kandalla dla wielu zmiennych.

REGRESJI - obliczenie współczynników regresji wielomianowej między zmiennymi niezależnymi i zmienną zależną.

Uwaga 16

W opisie rozkazu WSPOLCZYNNIK i dalszych pominięto spycyfikację list. Listy w tych rozkazach zawierają informację o danych i wynikach. W pracy tej świadomie pominięto dokładne spycyfikacje list. Spycyfikacje te będą podane w szczegółowym opisie języka JESTA.

Rozkaz POCZATKOWEOBLICZENIA

Format ogólny:

<u>POCZATKOWEOBLICZENIA,</u>	<u>SUMA</u> , lista;
	<u>WARTOSC</u> , lista;
	<u>ODCHYLENIE</u> , lista;
	<u>ROZKLAD</u> , lista;

Poszczególne słowa problemowe odpowiednio oznaczają:

SUMA - obliczenie sumy zmiennej losowej.

WARTOSC - obliczenie wartości średniej zmiennej losowej oraz wartości najmniejszej i największej tej zmiennej.

ODCHYLENIE - obliczenie odchylenia standardowego zmiennej losowej.

ROZKLAD - obliczenie ile wartości zaobserwowanej zmiennej losowej jest większych od danej wartości, ile jest mniejszych od drugiej danej wartości i ile jest między nimi.

Rozkaz STATYSTYKA

Format ogólny:

STATYSTYKA,

{ CHIKWADRAT,lista;
FRIEDMANA,lista;
QCOCHRANA,lista

Poszczególne słowa problemowe odpowiednio oznaczają:

CHIKWADRAT - obliczenie statystyki chi-kwadrat Pearsona dla tablic wielopolowych.

FRIEDMANA - obliczenie statystyki Friedmana dla zbadania czy pewna liczba prób pochodzi z tej samej populacji.

QCOCHRANA - obliczenie statystyki Q - Cochran dla analizy trzech lub więcej danych dychotomicznych.

Rozkaz KOWARIANCJA

Format ogólny:

KOWARIANCJA,

{ AUTO,lista;
KROS,lista;

Poszczególne słowa problemowe odpowiednio oznaczają:

AUTO - obliczanie autokowariancji szeregu chronologicznego dla opóźnień od 0 do L - 1.

KROS - obliczenie kroskowanacji szeregu A z szeregiem B wyprzedzającym A lub wyprzedzanym przez A.

Rozkaz WARIANCJA

Format ogólny:

WARIANCJA,lista;

Rozkaz ten dokonuje analizy wariancji na podstawie próbki o określonej liczności z rozbićciem na liczbę poziomów obserwacji według każdego z k danych kryteriów.

Rozkaz PRAWDOPODOBIENSTWO

Format ogólny:

PRAWDOPODOBIENSTWO,lista;

Rozkaz ten służy do obliczenia prawdopodobieństwa $P(x)$, że zmienna losowa X przyjmuje wartość mniejszą lub równą X, przy założeniu, że zmienna losowa X ma rozkład normalny (0,1).

Rozkaz NORMOWANIE

Format ogólny:

NORMOWANIE,lista;

Rozkaz ten służy do przeprowadzenia standaryzacji zmiennej losowej o rozkładzie normalnym $/M,5/$.

Rozkaz TESTOWANIEHIPOTEZY

Format ogólny:

TESTOWANIEHIPOTEZY

SA,lista;
SR,lista;
SN,lista;
SB,lista;
KORELACJI,lista;
GRUPOBSERWACJI,lista;

Poszczególne słowa problemowe odpowiednio oznaczają:

- SA - testowanie hipotezy, że średnia populacji o rozkładzie normalnym równa jest danej liczbie.
- SR - testowanie hipotezy, że średnie populacji A i B o rozkładach normalnych są równe przy założeniu, że wariancje ich są równe.
- SN - testowanie hipotezy, że średnie populacji A i B o rozkładach normalnych są równe przy założeniu, że wariancje ich są różne.
- SB - testowanie hipotezy, że średnie populacji A i B o rozkładach normalnych są równe przy założeniu, że brak jest informacji o ich wariancjach.
- KORELACJI - testowanie korelacji między dwoma zmiennymi losowymi za pomocą współczynnika Kendalla.
- GRUPOBSERWACJI - testowanie hipotezy, że dwie niezależne grupy obserwacji pochodzą z tej samej populacji.

Rozkaz KONIEC

Format ogólny:

KONIEC;

Jest to ostatni rozkaz programu oznaczający jego koniec i musi zawsze wystąpić.

Uwagi końcowe

Przystępując do prac nad realizacją przedstawionego języka problemowo ukierunkowanego planuje się osiągnąć trzy zasadnicze i równorzędne cele:

- zrealizowanie języka ukierunkowanego problemowo, który będzie praktycznym i wygodnym narzędziem dla użytkownika,
- zebranie doświadczeń dla opracowania metody generacji podobnych języków ukierunkowanych problemowo,
- wypracowanie mechanizmów, które po wbudowaniu do opracowanego języka pozwolą na proste rozszerzanie go o dodatkowe rozkazy.

Przedstawiony opis zawiera bardzo krótką i zwięzłą, a przez to niepełną charakterystykę realizowanego języka. Brak w nim wielu istotnych szczegółów, które celowo pominięto aby nie zaciemniały podstawowych cech języka. Opis ten nie jest także opisem w ujęciu matematycznym. Stanowi raczej opis języka przeznaczony dla jego przyszłych użytkowników, którzy nie są informatykami.

Literatura

BONKOWICZ-SITTAUER S., GRODZICKI S.: Oprogramowanie biblioteczne minikomputera MERA 400. Biuletyn Informacyjny OSK 1982 nr 1

Biuletyn Informacyjny NAUKI I TECHNIKI KOMPUTEROWE

mgr Grazyna WASIL

Instytut Maszyn Matematycznych

Zwięzłe listowanie zbiorów źródłowych

W okresie paroletniej pracy na minikomputerze MERA 400 napotymano pewne trudności związane z wykorzystywaniem systemu operacyjnego SOM-3; w Pracowni Oprogramowania Zastosowań IMM podejmowano różne próby pokonywania tych trudności. Opracowanie to jest przykładem jednego z takich rozwiązań.

Opracowano mianowicie program o nazwie "CO", który pozwala listować w dowolnie krótkiej formie, a więc oszczędzając papieru drukarki, zbiory źródłowe przechowywane na dysku. W ten sposób stworzono możliwość szybkiej kontroli zawartości dysku.

Prezentowane opracowanie rozpoczyna się omówieniem tych aspektów oprogramowania podstawowego maszyny MERA 400, z których wynika konieczność sporządzenia i dołączenia do oprogramowania systemowego programu "CO". Następnie będzie scharakteryzowany sam program i sposoby jego stosowania. Artykuł kończy się wnioskami sugerującymi metodę postępowania przy posługiwaniu się niedoskonałym oprogramowaniem.

Charakterystyka wybranych aspektów oprogramowania MERA 400

System operacyjny SOM-3 (bez procedury systemowej FMC) obsługujący maszynę MERA 400 w konfiguracji z jednym dyskiem nie zawiera podsystemu zbiorów, w związku z czym użytkownik ma poważne trudności z wykorzystaniem swoich zbiorów. Istota tych trudności polega na tym, że wszystkie funkcje podsystemu zbiorów użytkownik musi wykonywać sam, między innymi musi np. notować ręcznie na papierze: gdzie na dysku przechowuje swoje zbiory, ich nazwy, wersje, itp.; rzecz jasna nie może tych notatek zgubić.

Odczytywanie zawartości dysku też jest kłopotliwe, szczególnie nie można w łatwy sposób odczytać poprzeplatanych zbiorów postaci binarnej i źródłowej. Zbiory postaci źródłowej daje się odczytać tylko procedurą systemową UPD, a postaci binarnej tylko procedurą systemową LIB. Natomiast próba odczytania zbiorów takiej i takiej postaci kończy się wpadnięciem zastosowanej procedury (UPD lub LIB) w stan "run" i użytkownik tę systemową procedurę (UPD lub LIB) musi "abortować".

Ponadto odczytywanie zawartości dysku jest kłopotliwe ze względu na obazerność uzyskiwanych listingów. W szczególności dotyczy to listingu zbiorów źródłowych (dla zbiorów binarnych możliwość zwięzłego listingu daje dyrektywa LIST w LIB w trybie LNA). Trudności w uzyskiwaniu zawartości listingu zbiorów źródłowych polegają na tym, że używając UPD (dyrektywa LIST) w trybie LNA

otrzymujemy ciąg napisów wyszczególniający ile linii liczy zbiór źródłowy (bez nazwy tych zbiorów, a nawet bez numeracji zbiorów). Natomiast używając UPD w trybie LAL otrzymujemy pełne listingi zbiorów, każdy rozpoczynający się od nowej strony!

Program "CO" może spełniać funkcje przedstawiania katalogu zbiorów źródłowych sekcji dyskowej, tzn. może podawać początkowe linie zbiorów źródłowych, informacje o liczbie linii tych zbiorów oraz może sygnalizować występowanie zbiorów binarnych, nie źródłowych.

Koncepcja programu "CO"

Program "CO" nie nakłada żadnych wymagań wobec postaci zbiorów źródłowych, ale opiera się na założeniu, że zbiory źródłowe często zawierają treści programów, czy też podprogramów. Nazwy tych programów, podprogramów występujące na początku zbiorów źródłowych w jakimś stopniu identyfikują zbiór źródłowy. Tym niemniej wygodnie jest w programach lub podprogramach umieszczać komentarz, który jednoznacznie identyfikowałby zbiór (np. komentarz mówiący, która to jest wersja programu).

Program "CO" jest napisany w języku assembler. Daje to tę korzyść, że użytkownik wykonując program "CO" nie musi wcześniej robić dowiązań urządzeń zewnętrznych (np. drukarki).

Program "CO" korzysta ze standardowego podprogramu DCOMP - dekompresującego zbiory.

Sposób i przykłady użytkowania programu "CO"

Program "CO" dokonuje przeglądu podanej sekcji dyskowej wg zadanych parametrów i daje zwięzły listing, tzn:

- odlicza każde wystąpienie znacznika końca zbioru EOF ***** "NR"
- dla zbiorów, będących produktami "UPD", drukuje początkowe linie i informacje ile rekordów liczy zbiór/ ----> "LICZBA REKORDOW"/
- dla zbiorów, będących produktami "LIB" (TRYB: "GCIV"), "EOI", sygnalizuje wystąpienie zbioru binarnego/ZBIOR BINARNY - "NAZWA"/
- dla zbiorów, nie będących produktami "UPD", "LIB", "EOI", sygnalizuje występowanie resztek jakiegoś zbioru/RESZTKI ZBIORU/.

Stosowanie programu "CO" przebiega w sposób następujący: po pojawieniu się na konsoli komunikatu: "PODAJ:SEKCJA,CS-CSL,NRZB1,NRZB2,IELIN" należy w jednej linii (oddzielając przecinkami) podać:

- nazwę sekcji ze zbiorami źródłowymi,
- parametr "CS" lub "CSL", tj. zdecydować czy wydruk ma być skierowany na konsolę czy na drukarkę,
- liczbę reprezentującą pierwszy drukowany zbiór,
- liczbę reprezentującą ostatni drukowany zbiór,
- liczbę określającą ile początkowych linii każdego zbioru będzie drukowanych.

Jeżeli natomiast zamiast tych wszystkich parametrów zostanie podane EXI spowoduje to wyjście z programu "CO".

Po wprowadzeniu parametrów niezgodnie z podanymi w niniejszym opracowaniu regułami lub po zakończeniu listingu dla zadanej sekcji program wraca do stanu oczekiwania na parametry.

Parametr "SEKCJA" może przyjmować następujące wartości: AL1,AL2,AL3,AL4,AM1,AM2,AM3,AM4,AD1,AD2,AD3,AD4,ASA,ASB,ASC,ASW,WRK,AJC,AJW.

Program "CO" może generować następujące komunikaty:

- ZLA NAZWA SEKCJI
- ZLA NAZWA DRUKARKI
- NRZB - UJEMNY LUB ZEROWY
- NRZB1 > NRZB2
- ILELIN - UJEMNY LUB ZEROWY
- ZA MALO PARAMETROW
- NRZB1,NRZB2,ILELIN - MUSZA BYC LICZBAMI
- LICZBA ZBIOROW "NR" < NRZB1
- NAZWA SEKCJI LUB DRUKARKI ZAWIERA ZNAKI NIEDOPUSZCZALNE
- LICZBY MUSZA BYC CALKOWITE (MAX.6 ZNAKOW)
- KONIEC SEKCJI
- BLAD TRANSMISJI

Program "CO" służy, jak podano wyżej, do listowania początkowych linii zbiorów źródłowych. Natomiast informacje o zbiorach binarnych należy uzyskiwać za pomocą procedury systemowej "LIB". Niemniej program "CO" sygnalizuje występowanie takich zbiorów.

Dla ilustracji pracy programu "CO" pokazano na rys. 1 i 2 przykłady wydruku z konsoli (CS) i odpowiadający im wydruk z drukarki (CSL).

```

CS
UWAGA! WYJSCIE Z 'CO' SLOWEM 'EXI'
PODAJ: SEKCJE,CS-CSL,NRZB1,NRZB2,ILELIN
ASA CSL 1 8 3

PODAJ: SEKCJE,CS-CSL,NRZB1,NRZB2,ILELIN
ASA CSL 14 15 3
PODAJ: SEKCJE,CS-CSL,NRZB1,NRZB2,ILELIN
AL3 CSL 60 62 3
!JOB(CO ) BLAD TRANSMISJI
PODAJ: SEKCJE,CS-CSL,NRZB1,NRZB2,ILELIN
ALA
!JOB(CO ) ZA MALO PARAMETROW
PODAJ: SEKCJE,CS-CSL,NRZB1,NRZB2,ILELIN
ALA MA KOTA
!JOB(CO ) NRZB1,NRZB2,ILELIN -- MUSZA BYC LICZBAMI
!JOB(CO ) ZLA NAZWA DRUKARKI
PODAJ: SEKCJE,CS-CSL,NRZB1,NRZB2,ILELIN
EXI
*
```

Rys. 1. Przykład wydruku z konsoli (CS) podczas pracy programu "CO"

```

SEKCJA --- ASA
=====
*****          1
C*****
C   FOF2 (X,VAL,FCT,XST,EPS,IEND,IER)
C*****
*****          2
C*****
C   FOF3 (X,F,DERF,FCT,XST,EPS,IEND,IER)
C*****
*****          3
ZBIOR BINARNY --- PUT
*****          4
ZBIOR BINARNY --- GET
*****          5
RESZTKI ZBIORU
*****          6
*****          7
*****          8
          PGM II
*   TEST - ZNAJDOWANIE NAZWY W TABLICY
*   (1.2,BRUM,1,1.2.3,-KOT)
          -----)  82
          -----)  79
          -----)  0
          -----)  0
          -----) 138

SEKCJA --- ASA
=====
*****          14
*****          15
RESZTKI ZBIORU

SEKCJA --- AL3
=====
BLAD TRANSHISJI

```

Rys. 2. Przykład wydruku wyników pracy programu "CO" wyprowadzany na drukarkę (CSL)

Używając programu "CO" należy kierować się poniższymi zasadami przy doborze parametrów.

- Jeśli orientujemy się ile mamy zbiorów na sekcji dyskowej, to jako NRZB2 podajemy tę wielkość, jako NRZB1 podajemy liczbę 1 i wówczas otrzymujemy listing początkowych linii zbiorów umieszczonych od początku sekcji.
- Jeśli nie orientujemy się ile mamy zbiorów na sekcji dyskowej, to jako NRZB2 podajemy bardzo dużą liczbę (np. 1000), jako NRZB1 podajemy liczbę 1 i wówczas otrzymujemy listing początkowych linii zbiorów umieszczonych od początku sekcji oraz napis "KONIEC SEKCJI" (oznaczający, że cała sekcja została przejrzana).
- Jeżeli chcemy wyprowadzić informację o zbiorach od kolejnego zbioru sekcji, to jako NRZB1 podajemy aktualny numer tego zbioru na listowanej sekcji.
- Jeśli jako NRZB1 podajemy liczbę większą niż liczba zbiorów na listowanej sekcji dyskowej, to otrzymamy komunikat:
" LICZBA ZBIOROW N < NRZB1 " - gdzie N jest faktyczną liczbą zbiorów na sekcji.
- Jako ILELIN podajemy ile linii początkowych każdego zbioru chcemy drukować (zwykle tyle aby zidentyfikować zbiór, tj. aby linie te zawierały nazwę programu i komentarz). Jeśli ILELIN będzie miało wartość większą od liczby określającej najdłuższy zbiór na sekcji dyskowej, to działanie programu "CO" będzie się sprowadzało do listowania całych zbiorów i ewentualnie do sygnalizacji istnienia na dysku zbiorów binarnych.

- Każdy zbiór na dysku rozpoznawalny jest przez system operacyjny SOM-3 po znaczniku końca zbioru (tzw. EOF). Dostęp do zbioru możliwy jest po przewinięciu o żadaną liczbę znaczników EOF. Z tego powodu każde znalezienie przez program "CO" znacznika EOF na sekcji dyskowej jest traktowane jako obliczanie zbioru. Jeżeli jest to zbiór pusty, to na wydruku jest sygnalizowany tekstem: ----> 0 .

Wnioski

Program "CO" wraz z procedurą systemową "LIB" jest pomocny w szybkim określaniu zawartości dysku, jest to szczególnie ważne gdy w systemie brakuje katalogu zbiorów.

Tak więc przedstawiony program "CO" uzupełnia w pewnym stopniu luki oprogramowania. Wprawdzie użytkownikowi nadal nie będzie łatwo korzystać ze swoich zbiorów, ale przynajmniej będzie mógł łatwo listować zawartość dysku.

Program "CO" listuje zawartość sekcji dyskowej, tzn. wszystko co kiedykolwiek na niej zostało zapisane. Powstała więc konieczność dołączenia do standardowego oprogramowania SOM-3 następnego programu "CZYSCI", który wymazywałby zbędne dla użytkownika zbiory z sekcji.

Tworzenie dodatkowych programów ułatwiających korzystanie z MERA 400 w ramach obecnej wersji systemu SOM-3 jest jedną z metod posługiwania się maszyną. Jest to droga łatwiejsza niż pisanie systemów dla własnych potrzeb, a tylko nieco trudniejsza od korzystania z istniejących możliwości systemu, jak np. używania procedur JOB CONTROL-a, którymi można również prowadzić kontrolę dostępu do sekcji dyskowej.

Literatura

- [1] System SOM-3: MERA 400. Dokumentacja techniczno-ruchowa. Maszynopis.
- [2] Madnick S.E., Donovan J.J.: Operating Systems. McGraw Hill 1974.

Sprawozdania z konferencji

SZKOŁA METODOLOGII KONSTRUOWANIA MASZYN

Zorganizowana w tym roku szkoła jest kontynuacją zeszłorocznej inicjatywy (Bierutówce - zob. sprawozdanie dr inż. R.Pawlika w numerze 5-6/1981 BIOSK).

Skupione w Rydzynie grono pracowników naukowych i konstruktorów z wyższych uczelni, instytutów, OBR-ów i biur konstrukcyjnych niezmiernie intensywnie (zajęcia przed i po południu) dyskutowało i analizowało zarówno ogólne zasady twórczego podejścia do problemów projektowania i konstruowania jak i bardziej konkretne, związane z realnymi warunkami, sposoby ich rozwiązywania.

Zasadnicza tematyka Szkoły była prowadzona w czterech głównych ujęciach: przede wszystkim przez doc.dr hab. R.Rohatyńskiego z Politechniki Wrocławskiej (część II materiałów), przy czym na zajęciach seminaryjnych problemy te były rozwijane przez ww oraz dr Z.Chrostowskiego.

Podobne nieco, aczkolwiek przepojone własną specyfiką, były materiały prezentowane przez prof. W.Lankiewicza i doc. B.Machowskiego (część III materiałów z Krakowskiej AGH). Odmienny sposób podejścia do problematyki doboru cech konstrukcyjnych, ale realizowany z wzorową jasnością prezentowania materiałów, przedstawił doc. Z.Jaskoła z Politechniki Śląskiej (materiały - część IV). Wreszcie optymalizacyjne techniki dla projektowania omówił szeroko i barwnie doc. W.Tarnowski też z Politechniki Śląskiej (materiały - część V). Jedyne z zaplanowanych wykładów i seminarium "z przemysłu" pt. "Praktyczne metody oceny dla porównania rozwiązań przez określenie poziomu jakości" - został przeprowadzony przez mgr inż. S.Drzemalę z OBR BUMAR - Łabędy (konspekt w materiałach). Przykład ten był doskonałą ilustracją podnoszonych przez licznych uczestników Szkoły rozbieżności między metodami opracowywanymi przez naukowców a konkretyzacją tych metod w praktyce. Przy czym szczególnie ciekawe głosy w dyskusji padały z ust przedstawicieli: Biura Konstrukcyjnego z Wrocławskiej FADROM-y, z Zakładów Autobusowych w Jelczu, czy Zakładu Konstrukcji z Tworzyw Sztucznych w Poznaniu.

Należy też podkreślić, że wśród tych wszystkich, w miarę typowych wypowiedzi, szczególny wpływ na przebieg dyskusji, jej ton i poziom miała obecność w Rydzynie prof. dr hab. Stefana Ziemby z IPPT-PAN. Zarówno jego planowe wykłady (część I materiałów), jak i niezliczone rozmowy kulturalne i między zajęciami, a wreszcie wypowiedzi w trakcie "obrad okrągłego stołu" podnosiły poziom dyskusji i mobilizowały innych do intensywnego myślenia.

Ogólny dobry nastrój Szkoły bez wątpienia w dużej mierze osiągnięto dzięki staraniom organizatorów, ulokowaniu Szkoły w pięknym zamku w Rydzynie, który jest obiektem przystosowanym do tego typu działalności - (sale audiowizualne, sale kulturalne), porządne pokoje hotelowe, przyzwoite wyżywienie i brak komunikacji z większym miastem, powodowały wyizolowanie się ze spraw bieżących a skupienie się na dyskutowanych problemach.

OFERTA

Laboratorium Grafiki Komputerowej IMM

poleca swoje usługi w zakresie wykonywania prac analityczno-programistycznych na minikomputerze MERA 400, z wykorzystaniem standardowych urządzeń wejścia/wyjścia oraz następujących urządzeń graficznych:

- rastrowego monitora graficznego
MERA 7954

- ploterów: Xinetios
Benson
Digigraf
KL-2

- Dysponujemy - poza systemem standardowym - kompletnym systemem graficznym PSG do programowania na minikomputerze MERA 400, sposobem wsadowym lub konwersacyjnym, dowolnych wyjściowych urządzeń graficznych.
- Udzielamy konsultacji z dziedziny grafiki komputerowej i metod numerycznych.
- Przyjmujemy zlecenia na zaprojektowanie i uruchomienie kompletnych systemów graficznych lub ich fragmentów, a nawet opracowanie poszczególnych algorytmów, w szczególności w zakresie wspomagania projektowania.

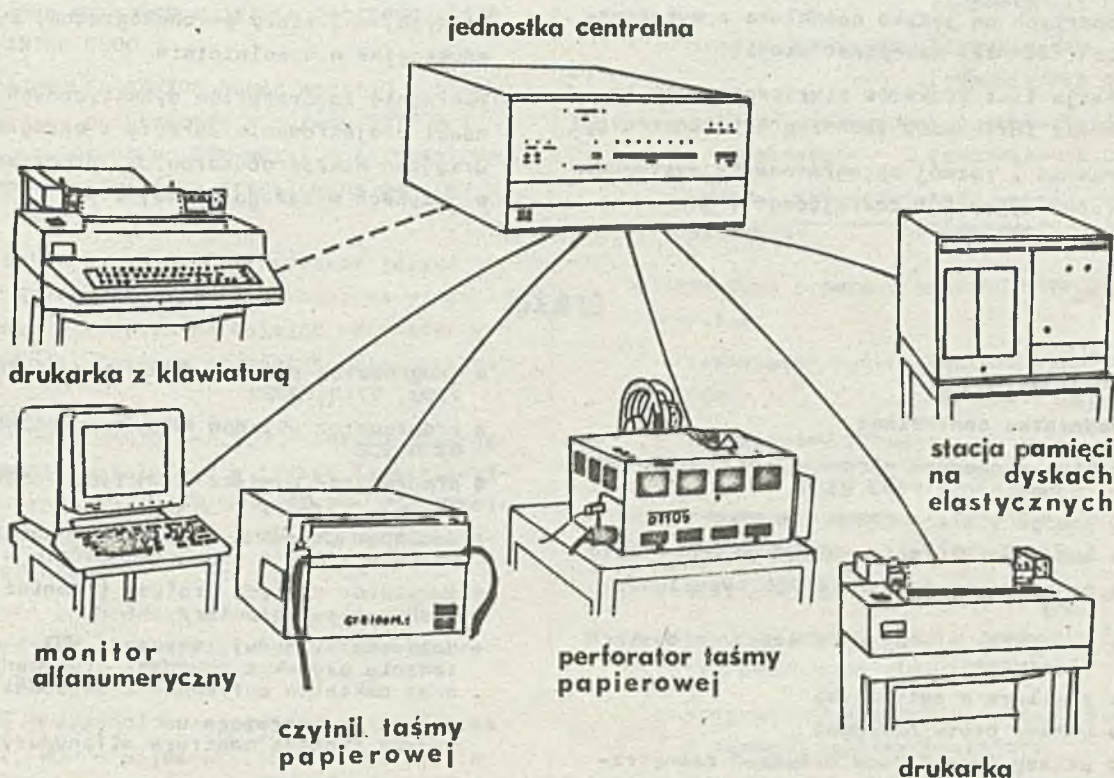
Bliższe informacje można uzyskać w Pracowni
Grafiki Komputerowej Instytutu Maszyn Matemat.
ul. Krzywickiego 34, 02-078 Warszawa
tel. 21-84-41 w. 428 lub 396



OFERTA

ZESTAW SPRZĘTU I OPROGRAMOWANIA DO PROJEKTOWANIA, URUCHAMIANIA I TESTOWANIA URZĄDZEŃ MIKROPROCESOROWYCH

Mikroprocesorowy system wspomagania projektowania (MSWP)



MSWP jest wielofunkcyjnym narzędziem wspomagania prac projektowo-konstrukcyjnych przeznaczonym do opracowywania, uruchamiania i badania sprzętu oraz oprogramowania urządzeń mikroprocesorowych, realizowanych na układach serii K580 produkcji ZSRR lub odpowiednikach typu INTEL 8080 oraz na układach serii K589 produkcji ZSRR lub odpowiednikach typu INTEL 3000.

MSWP pod względem funkcjonalnym stanowi

odpowiednik takich systemów jak Intellec MDS 800 firmy Intel (USA) lub Microprocessor Lab. 8002 firmy Tektronix (USA).

Modułowa struktura funkcjonalna jednostki centralnej MSWP, wyposażona w systemowy interfejs typu MULTIBUS, z możliwością dołączania różnych urządzeń zewnętrznych, pozwala na kompletowanie zróżnicowanych konfiguracji dla konkretnych zastosowań i łatwą rozbudowę systemu.

ZASTOSOWANIA

1. Zastosowania MSWP do wspomagania projektowania urządzeń mikroprocesorowych
 - uruchamianie sprzętu urządzeń mikroprocesorowych z wykorzystaniem modułu emulatora układowego
 - opracowywanie oprogramowania użytkowego dla urządzeń mikroprocesorowych, a w szczególności:
 - edycja i kompilacja programów źródłowych z języka assemblera lub języków wyższego rzędu
 - uruchamianie programów wynikowych
 - poprawianie i symulacja mikroprogramów
 - tworzenie języków ukierunkowanych problemowo opartych na języku assemblera z wykorzystaniem techniki makroinstrukcji
 - emulacja list rozkazów mikroprocesorów innych niż INTEL 8080 techniką makroinstrukcji
 - tworzenie i rozwój oprogramowania systemowego (operacyjnego i testującego)
 - uruchamianie urządzeń mikroprocesorowych za pomocą emulatora układowego
 - testowanie obwodów drukowanych, układów scalonych i obwodów z układami scalonymi
 - programowanie pamięci EPROM, PROM i układów FPLA
2. Inne przykłady zastosowania MSWP
 - sterowanie i przetwarzanie informacji w systemach kontrolno-pomiarowych
 - zbieranie i przetwarzanie danych w systemach handlowych, bibliotecznych, medycznych, itp.
 - tworzenie i wykorzystanie systemów konwersacyjnych, np. testy psychologiczne, systemy edukacyjne w szkolnictwie
 - tworzenie laboratoriów dydaktycznych np. do nauki projektowania sprzętu i oprogramowania urządzeń mikroprocesorowych, programowania w językach wyższego rzędu, itp.

SPRZĘT

1. Zestaw podstawowy
 - 1.1. Jednostka centralna:
 - blok procesora głównego typu INTEL 8080 odpowiednik K580 MK80
 - pamięć stała - EPROM lub PROM - 4kB
 - pamięć o dostępie swobodnym RAM - 32kB
 - programator pamięci EPROM typu INTEL 1702
 - jednostka sterująca pamięci na dyskach elastycznych
 - zasilacz z automatyką
 - układy płyty czołowej
 - układy interfejsów urządzeń zewnętrznych
 - 1.2. Urządzenia zewnętrzne:
 - stacja pamięci na dyskach elastycznych (8 cali z jednostronnym zapisem)
 - konsola systemowa (monitor alfanumeryczny z klawiaturą wyposażony w interfejs V24 lub monitor TV, lub drukarka z klawiaturą DZM 180 11A KSR)
 - czytnik taśmy perforowanej CT 2100 lub CT 1000
 - perforator taśmy DT 105S lub DT 105
2. Moduły opcjonalne jednostki centralnej
 - pamięć stała (moduły 4kB) do przechowywania stałych programów systemowych
 - pamięć o dostępie swobodnym (moduły 8kB z możliwością kompletowania do 64kB)
 - programator pamięci EPROM typu INTEL 2704, 2708, 2716, 2732
 - programator układów FPLA typu SIGNETICS 82 S 100*
 - programator pamięci PROM typu TUNGSRAM TM 621, TM 622, TM 624*
 - emulator układowy dla mikroprocesorów typu INTEL 8080 i jego odpowiedników
 - symulator pamięci stałych (również szybkich pamięci bipolarnych)
 - uniwersalny moduł testujący UMT-1 do testowania płytek z obwodami drukowanymi oraz pakietów cyfrowych z układami TTL*
 - jednostka sterująca odbiornikiem TV realizująca funkcje monitora alfanumerycznego.
3. Urządzenia dodatkowe
 - klawiatura alfanumeryczna KL-1
 - lampa kasująca zapis pamięci EPROM promieniowaniem ultrafioletowym LK-1
 - dodatkowe urządzenia zewnętrzne:
 - drukarka DZM 180 z interfejsem równoległym,
 - teletajp jako uniwersalne urządzenie wejścia/wyjścia.

* dostawa w 1984 r.

OPROGRAMOWANIE

1. Oprogramowanie na taśmach perforowanych

- MONITOR - podstawowy system operacyjny wbudowany w sprzęt MSWP (w pamięci PROM) - w kilku wersjach
- EDYTOR 1 wer.1.6 (REEDYT) - edytor tekstów z buforem tekstu wielkości ograniczonej rozmiarem pamięci RAM
- EDYTOR 2 wer.1.0 - edytor tekstów zapisanych na taśmie perforowanej, wykorzystujący bufor tekstu wielkości równej liczbie znaków poprawianego wiersza tekstu
- MAKROASEMBLER REM wer.1.4 - translator programów pisanych w języku makroasemblera (typu INTEL 8080)
- MAKROASEMBLER-EDYTOR REMAK wer.1.0 - program będący połączeniem programu EDYTOR 1 wer.1.6 i programu REM wer.1.4 ze wspólnym buforem zawierającym translowany program źródłowy
- MINI BASIC wer.1.0 - interpreter języka BASIC działający w trybie konwersacyjnym z krokowym wykonywaniem kolejno wprowadzanych instrukcji. Zawiera arytmetykę stałoprzecinkową
- INTERPRETER BASIC wer.1.0 - interpreter języka BASIC działający w trybie konwersacyjnym z 6kB buforem przechowującym instrukcje i dane programu. Zawiera arytmetykę zmienoprzecinkową
- EMULATOR 8080 wer.3.0 - oprogramowanie modułu emulatora dla systemów z mikroprocesorem typu INTEL 8080
- SYMROM wer.2.0 - oprogramowanie symulatora pamięci ROM o pojemności 512x64 bitów dla systemów z mikroprocesorem typu INTEL 3000
- TEST RAM - zestaw testów pamięci RAM
- ZESTAW TESTÓW CZYTNIKA CT 2100 I PERFORATORA DT 1055
- MONITOR TESTERA UMT-1 wer.1.1 - program realizujący wczytanie z taśmy perforowanej testy pakietów na testerze UMT-1
- JEZYK TESTERA UMT-1 wer.1.1 - translator języka testera UMT-1 na kod sterujący

2. Dyskowe systemy operacyjne i programy działające pod nadzorem tych systemów na dyskach elastycznych

2.1. OS-I wer.1.4 (odpowiednik CP/M)

- podstawowy zestaw programów systemu MSWP OS-I wer.1.4:
 - kontekstowy edytor tekstów (odpowiednik ED)
 - translator asemblera typu INTEL 8080 (odpowiednik ASM)
 - program transferowy (odpowiednik PIP)
 - debugger (odpowiednik DDT)
 - program zmieniający format zbioru (odpowiednik LOAD)
 - program statusowo-statystyczny (odpowiednik STAT)
 - program wyprowadzający zawartość zbioru dyskowego (odpowiednik DUMP)
 - program interpretujący makrodyrektywy systemowe (odpowiednik SUBMIT)
 - dodatkowe programy systemu MSWP OS-I wer.1.4:
 - translator makroassemblera typu INTEL 8080 (odpowiednik MAC)
 - symboliczny debugger (odpowiednik SID)
 - program formatujący tekst wg reguł edytorskich (odpowiednik TEX)
 - test pamięci na dyskach elastycznych
 - program odzyskujący z dokładnością 1kB uszkodzone zbiory
 - program odzyskujący z dokładnością 1 rekordu uszkodzone zbiory
 - program uzdatniający uszkodzoną dyskietkę
 - program sprawdzający kopię taśmową zbioru dyskowego
- dyskowe wersje programów testujących:
- ZESTAW TESTÓW PAMIĘCI RAM wer.2.1
 - ZESTAW TESTÓW CZYTNIKA CT 2100 I PERFORATORA DT 1055 wer.2.1
 - MONITOR TESTERA UMT-1 wer.2.1
 - JEZYK TESTERA UMT-1 wer.2.1

2.2. OS-II ver.2.0 (odpowiednik ISIS-II)

- podstawowy zestaw programów systemu OS-II ver.2.0:

program formatowania dyskietek i kopiowania zbiorów (odpowiednik FORMAT)

program interpretujący makrodyrektywy systemowe (odpowiednik SUBMIT)

program wyświetlający zawartość katalogu (odpowiednik DIR)

program kopiujący zbiory (odpowiednik COPY)

program usuwający zbiory (odpowiednik DELETE)

program zmieniający atrybuty zbiorów (odpowiednik ATTRIB)

programy zmieniające formaty zbiorów
odpowiedniki BINOBJ, HEXOBJ, OBJHEX

program zarządzający bibliotekami
(odpowiednik LIB)

program przekształcający relokowalne zbiory wynikowe na zbiory wynikowe o adresach absolutnych
(odpowiednik LOCATE)

program łączący moduły wynikowe
(odpowiednik LINK)

- dodatkowe programy systemu OS-II ver.2.0:

edytor tekstów (odpowiednik EDIT)

translator makroassemblera relokowalnego typu INTEL 8080
(odpowiednik ASM-80)

kompilator języka PL/M
(odpowiednik PL/M-80)

Opracowanie: Instytut Maszyn Matematycznych,
Krzywickiego 34, 02-078 Warszawa, Polska
telefon 218441, telex 813517

Producent: Zakład Doświadczalny IMM,
Krzywickiego 34, 02-078 Warszawa, Polska
telefon 218441, telex 813517

Eksporter: METRONEX
ul. Mysia 2, 00-950 Warszawa
telefon 21-41-81, telex 814471

OFERTA

Analizator lotu głowicy pamięci dyskowej

Urządzenie służy do obserwacji na ekranie monitora TV usytuowania stopki głowicy w stosunku do wirującego dysku i umożliwia dokładne pomiary odległości poszczególnych punktów stopki od dysku. Działanie urządzenia jest oparte na wykorzystaniu interferencji światła, którą uzyskuje się przez oświetlenie światłem monochromatycznym głowicy umieszczonej pod dyskiem szklanym. Układ prążków interferencyjnych, pojawiający się na ekranie na tle obrazu głowicy, pozwala na określenie wielu parametrów lotu (np. punktu minimalnego zbliżenia, pochylecia głowicy i in.). Pomiaru bezwzględnej odległości dokonuje się przez zmianę długości fali w taki sposób, aby uzyskać określony przesuw prążków w wybranym punkcie stopki, zaznaczonym na ekranie za pomocą znacznika.

Analizator składa się z modułu opto-mechanicznego, zawierającego zaadaptowaną dla urządzenia pamięć dyskową, monochromator, układ optyczny i kamerę TV, oraz modułu elektroniki z monitorem i układami elektronicznymi, umożliwiającymi obliczenie odległości głowicy od dysku, generację i przesuw znacznika, wybieranie stref położenia badanej głowicy i wprowadzenie danych dotyczących badanej głowicy do wydruku.

Analizator stosuje się do badania głowicy zarówno na etapie ich opracowywania, jak i kontroli jakości głowic przy produkcji seryjnej.

Główne parametry

● Zakres mierzonych odległości głowicy od dysku	0,6 + 2,5 μ m
● Prędkość obrotowa dysku	2400 obr./min
● Liczba stref położenia głowic	3
● Dokładność pomiaru odległości głowicy od dysku	\pm 10%
● Zasilanie	220 V, 50 Hz, 700 VA
● Wymiary	moduł opto-mechaniczny: 1230 mm x 580 mm x 1335 mm moduł elektroniki: 515 mm x 705 mm x 1310 mm

Informacji szczegółowych udziela:

Kierownik Zakładu Technologii i Aparatury IMM
mgr inż. Eugeniusz NOWAK
tel. 21-84-41 w. 287

lub

Kierownik Pracowni Urządzeń Zewnętrznych i Aparatury IMM
mgr inż. Romuald SYNAK
tel. 21-84-41 w. 319

OFERTA

Akustooptyczny modulator światła MC 102

Modulator jest przeznaczony do modulacji impulsowej natężenia wiązki światła laserowego. Składa się z głowicy modulującej, działającej na zasadzie dyfrakcji Bragga w ośrodku z rozchodzącą się falą ultradźwiękową, oraz zespołu elektronicznego, służącego do sterowania głowicą sygnału w.o.z. modulowanym impulsowo. Wewnętrzny generator impulsów umożliwia wykorzystanie modulatora jako niezależnego urządzenia do generacji impulsów laserowych o regulowanym czasie trwania i okresie powtarzania. Możliwa jest również generacja pojedynczego impulsu świetlnego o zadanym czasie trwania.

Modulator może być zastosowany w drukarkach laserowych, urządzeniach laserowej obróbki materiałów, w badaniach materiałów i innych pracach wymagających użycia impulsów świetlnych o określonej szerokości i repetycji.

Główne parametry głowicy modulującej

● Zakres długości fal	440 + 633 nm
● Apertura	1,5 mm
● Wydajność dyfrakcyjna (przy 2 W, 633 nm)	70 %
● Kontrast	1000:1
● Częstotliwość środkowa pasma przepuszczenia	65 MHz
● Kątowa separacja między wiązkami wejściową i wyjściową (przy 633 nm)	11 mr
● Wymiary	57 mm x 68 mm x 22 mm

Główne parametry zespołu sterującego

● Rodzaje pracy	sterowanie zewnętrzne sterowanie wewnętrzne generacja pojedynczego impulsu
● Częstotliwość nośna	65 MHz
● Moc wyjściowa	max. 2 W przy obciążeniu 50 Ω (regulowana płynnie)
● Okres powtarzania impulsów	150 ns + 10 s (8 podzakresów z regulacją ciągłą)
● Czas trwania impulsu	100 ns + 10 s (8 podzakresów z regulacją ciągłą)
● Zewnętrzny sygnał sterujący	napięcie TTL, obciążalność 2 jedn. log.
● Wymiary	293 mm x 140 mm x 220 mm

Informacji szczegółowych udziela:

Kierownik Zakładu Technologii i Aparatury IMM
mgr inż. Eugeniusz NOWAK
tel. 21-84-41 w. 287

lub

Kierownik Pracowni Urządzeń Zewnętrznych i Aparatury IMM
mgr inż. Romuald SYNAK
tel. 21-84-41 w. 319

OFERTA

sprzedaży systemu EKTRAN

Instytut Maszyn Matematycznych oferuje do sprzedaży system EKTRAN

Krótki opis aktualnej wersji systemu EKTRAN zawiera załączona instrukcja. Szczegółowy opis systemu wypożyczyć można w IMM; informacji telefonicznych udziela mgr inż. Jerzy SWIANIEWICZ tel. 21-84-41 w.314

Cena aktualnej wersji systemu wynosi 200 tys. zł. Użytkownik otrzymuje:

- szczegółowy opis funkcjonalny systemu /3 egzemplarze/
- materiał programowy w postaci dystrybucyjnej, tj. biblioteki load-modułów i makroinstrukcji
- materiał programowy w postaci źródłowej /programy w języku ASSEMBLER/.

Wyżej podana cena nie obejmuje nadzoru autorskiego. Nabywcom systemu, którzy chcieliby skorzystać z takiego nadzoru proponujemy zawarcie umowy-zlecenia na okres pół roku na zasadzie ceny zryczałtowanej 50.000 zł za cały okres nadzoru.

Zakres nadzoru autorskiego obejmuje:

- pomoc przy wdrażaniu systemu EKTRAN
- gwarancja uzyskania nowych poprawionych i udoskonalonych wersji systemu w okresie trwania umowy
- możliwość zgłaszania udoskonalień i usprawnień systemu.

Jeżeli propozycje te zostaną uwzględnione zgłaszającemu zapewnia się otrzymanie nowej wersji systemu nawet wówczas, gdy zgłoszone zmiany zostaną wprowadzone po zakończeniu okresu nadzoru autorskiego.

W proponowaną cenę zakupu aktualnej wersji systemu i nadzoru autorskiego nie wchodzi opracowania rozszerzające system o nowe funkcje. Przewidujemy bowiem rozbudowę systemu. Każda z nowych funkcji będzie wyceniana oddzielnie i może być dołączona do systemu zgodnie z życzeniem użytkownika. Informacje w tej sprawie będą przesyłane użytkownikom na bieżąco.

Zainteresowanych zakupem systemu EKTRAN informujemy, że zamówienia należy przysyłać na adres:

Instytut Maszyn Matematycznych
• Dział Ekonomiczny i Planowania
ul. Krzywickiego 34
02-078 Warszawa

Instalacja systemu nastąpi w przeciągu 1-2 miesięcy od daty potwierdzenia zamówienia na nadzór autorski lub w terminie uzgodnionym przez zainteresowanych.

System EKTRAN dla maszyn IBM i JS

System EKTRAN jest systemem konwersacyjnym pracującym w ramach systemów operacyjnych OS/MVT, OS/MVT lub OS/VS1 i obsługującym monitory ekranowe typu IBM-3270 (w szczególności MERA-7910 produkowane przez MERA-ELZAB). Umożliwia on wykorzystanie powyższych urządzeń do realizacji następujących funkcji:

- uzyskiwanie wglądu do informacji zawartych na nośnikach maszynowych jak np. dyskach, taśmach magnetycznych, PAO,
- uzyskiwanie wglądu do informacji znajdujących się w kolejkach wyjściowych - SYSOUT - z możliwością przenoszenia wyników JOB-ów z kolejki bądź ich usuwania,
- poprawianie i aktualizowanie zbiorów tekstowych znajdujących się na dyskach,
- przekazywanie tekstów zawartych w zbiorach sekwencyjnych lub podzbiorach zbiorów bibliotecznych na wejściu do READER-a systemowego,
- wykonywanie różnych funkcji pomocniczych administratora systemu,
- dokonywanie prostych zmian na dyskach i w PAO.

Dostęp do operacji, za pomocą których można naruszyć integralność systemu jest ograniczony przez zainstalowane protekcje oparte na systemie kont i haseł.

Hierarchiczna i modułowa struktura systemu EKTRAN umożliwia rozwój tego systemu na dwóch poziomach.

Pierwszy poziom wyznaczony jest przez ścisły interfejs między programem sterującym (MONITOR) i programami realizującymi konwersację (PROCESORY). Możliwy jest rozwój systemu przez dołączanie nowych procesorów współpracujących z monitorem zgodnie z uzgodnionym interfejsem. Z kolei różne wersje programów sterujących (MONITOROW) umożliwiają pracę systemu EKTRAN w różnym otoczeniu systemowo-programowym. W szczególności istnieją monitory sterujące pracą systemu:

- samodzielnie, metodą dostępu BTAM,
- we współpracy z MCP/TCAM,
- pod kontrolą TSO,
- w ramach innych systemów monitorowo-ekranowych (jak np. system SK, opracowany w CO-MPM, system SME, opracowany w FSO lub MONITOR-TLEPRZETWARZANIA, opracowany w BOOIIPM "POLMO").

Drugi poziom stwarza specjalny procesor, tzw. formatujący. Praca tego procesora jest sterowana programami napisanymi w specjalnym języku FL. Język ten pozwala na programowanie wyszukiwania informacji w pamięciach (dyskach, taśmach, PAO), ich selekcję oraz formatowanie w celu wyświetlenia na ekranie monitora. Możliwości systemu EKTRAN mogą więc łatwo być zwiększane przez włączanie do niego nowych - napisanych w języku FL - programów formatujących tzw. FORMATOW.

Procesor formatujący stwarza również pewne środki bezpośredniego oddziaływania użytkownika monitora ekranowego na pracę programu formatującego. Tym samym, przy odpowiednim zaprogramowaniu Formatu - użytkownik może na bieżąco sterować wybieraniem informacji wyświetlanych na ekranie.

System EKTRAN znajduje zastosowanie jako mocne narzędzie programisty systemowego lub aplikacyjnego przy uruchamianiu, instalowaniu i konserwacji systemów programowych, jak również przy administrowaniu zbiorami (bazami danych). Może on być również z powodzeniem wykorzystywany jako narzędzie szkoleniowe, za pośrednictwem którego można łatwo konfrontować informacje teoretyczne na

temat struktur danych w maszynie z rzeczywistością oglądaną "na żywo".

Inne zastosowanie systemu EKTRAN wynika z możliwości bezpośredniej konwersacyjnej edycji tekstów i przekazywania ich jako JOB-ów do przetwarzania wsadowego. Daje to możliwość zorganizowania konwersacyjnego dostępu użytkowników do systemu pracującego w reżimie wsadowym.

2. Obecne możliwości systemu

W obecnym stadium rozwoju system EKTRAN współpracuje z następującymi procesorami:

- ERDA - realizujący funkcje wyświetlania zawartości dysków magnetycznych,
- ETAPE - realizujący funkcje wyświetlania zawartości taśm magnetycznych,
- EZAP - realizujący funkcje wyświetlania zawartości dysków i PAO oraz umożliwiający dokonywanie zmian na dyskach i w PAO,
- ETEXT - procesor formatujący, umożliwiający programowanie sposobu wyświetlania informacji umieszczonych w pamięciach maszyny dyskach, taśmach magnetycznych, PAO ,
- EUPTE - realizujący funkcje edytora tekstów umieszczonych na dysku w zbiorach sekwencyjnych lub bibliotecznych oraz pozwalający na przekazywanie tekstów na wejście do READER-a systemowego.

Trzy pierwsze z powyższych procesorów ERDA, ETAPE, EZAP są to tzw. procesory podstawowe. Umożliwiają one wyświetlanie dowolnych rekordów bloków dyskowych i taśmowych lub wskazanych obszarów PAO. Informacje wyświetlane są w postaci szesnastkowo-tekstowej postaci "dump". Wyświetlane rekordy należy lokalizować podając ich adresy absolutne. Procesor EZAP umożliwia ponadto dokonywanie elementarnych modyfikacji informacji zawartych na dyskach lub w PAO.

Procesor ETEXT jest procesorem formatującym, interpretującym programy tzw. FORMATY zapisane w języku FL.

Do chwili obecnej opracowano formaty realizujące następujące funkcje wglądu do informacji znajdujących się na nośnikach maszynowych:

- LFBC - oglądanie zbiorów dyskowych
- LDYSK - oglądanie zbiorów PS i PO na dysku
- LISAM - oglądanie zbiorów o organizacji IS
- LISK - oglądanie wskazanych rekordów zbioru IS
- LMCSEC - oglądanie zawartości modułów ładownych
- LVTOC - wyświetlanie zawartości VTOC
- LCAT - lokalizacja zbioru w katalogu
- LDSS - oglądanie struktury zbioru na dysku
- LTAPE - wyświetlanie zawartości taśmy magnetycznej
- LTMS - oglądanie struktury zbiorów na taśmie magnetycznej
- LPAO - wyświetlanie informacji z PAO
- LVSOUT - wyświetlanie zbiorów z kolejki SYSOUT w systemie VS1
- LOSOUT - wyświetlanie zbiorów z kolejki SYSOUT w systemach MFT/MVT
- LQSOUT - wyświetlanie zawartości kolejek SYSOUT MFT/MVT
- LPRMFT - wyświetlanie modułów załadowanych do PAO MFT
- LPRMVT - wyświetlanie modułów załadowanych do PAO MVT
- LPRVS1 - wyświetlanie modułów załadowanych do PAO VS1
- LSCRA - usuwanie zbiorów z dysku
- LSMEM - operacje na skorowidzu zbioru PO.

Krótką informację o funkcjach i parametrach wejściowych poszczególnych formatów można uzyskać

na ekranie przez wprowadzenie tekstu HELP jako nazwy formatu.

LFBC, LISAM, LISK, LTAPE, i LPAO są tzw. formatami otwartymi. Oznacza to, że liczba sposobów formatowania informacji wyświetlanej na ekranie może być dla tych formatów zwiększana przez dołączenie do biblioteki systemu EKTRAN odpowiednich tzw. podformatów. Wśród parametrów wejściowych formatu otwartego występuje zawsze parametr, za pomocą którego można wybrać właściwy podformat formatujący. Dla formatu LPAO funkcję takiego parametru wejściowego pełni parametr OPERACJA, zaś dla pozostałych formatów otwartych LFBC, LISAM, LISK i LTAPE funkcję tę spełnia parametr POSTAC.

Podformaty, podobnie jak formaty, programuje się w języku PL. Zaprogramowanie podformatu dla określonego formatu otwartego jest na ogół czynnością bardzo prostą, bowiem sprowadza się do określenia sposobu wyświetlania informacji, których odnalezienie, selekcja i sprowadzenie do PAO dokonane zostało wcześniej przez format.

Format LPAO zaopatrzony jest standardowo w następujący komplet podformatów:

- PAO - wyświetlanie zawartości PAO w postaci szesnastkowo-tekstowej
- TIT - wyświetlanie adresów DCB, DEB, UCB przyporządkowanych nazwom zdań DD występujących w TIOT dla wykonywanego JOB-STEP-u
- TCK - wyświetlanie listy adresów TCB w systemie z przyporządkowanymi nazwami JOB-ów i JOBSTEP-ów
- TMT - wyświetlanie "drzewa subtasków" dla danego tasku
- TCM - wyświetlanie informacji o konfiguracji systemu telekomunikacyjnego określonego przez działający program MCP - TCAM
- UNT - wyświetlanie adresów bloków UCB oraz charakterystyk urządzeń systemu
- TCB - wyświetlanie wskazanego bloku TCB
- DCB - wyświetlanie wskazanego bloku DCB
- DEB - wyświetlanie wskazanego bloku DEB
- TIO - wyświetlanie wskazanego bloku TIOT
- CR - wyświetlanie zawartości rejestrów sterujących IBM 370
- QUE - wyświetlanie zawartości kolejek wejściowych JOBQ i wyjściowych SQUITQ w systemie VS1
- EKR - wyświetlanie tablicy sterującej programem EKTRAN MONITOR dla systemu lokalnego, w szczególności wyświetlane są adresy PORO związanych z poszczególnymi ekranami
- POR - wyświetlanie PORO.

Procesor EUPTE umożliwia redagowanie tekstów programów, poprawianie ich, kompletowanie JOB-ów i przekazywanie ich do wykonania przez program READER. Umożliwia on zatem konwersacyjną obsługę systemu wsadowego, a więc spełnia funkcje analogiczne do funkcji CRJE.

Procesor EUPTE operuje na dowolnych zbiorach sekwencyjnych lub bibliotecznych o 80-bajtowych dokumentach. Aktualizacja tekstu dokonywana jest "w miejscu".

WARUNKI PRENUMERATY

Prenumeratę na kraj przyjmują Oddziały RSW "Prasa-Książka-Ruch" oraz urzędy pocztowe i doręczyciele w terminie do dnia 25 listopada na rok następny.

Cena prenumeraty rocznej zł 840.

Jednostki gospodarki uspołecznionej, instytucje, organizacje i wszelkiego rodzaju zakłady pracy zamawiają prenumeratę w miejscowych Oddziałach RSW "Prasa-Książka-Ruch", w miejscowościach zaś, w których nie ma Oddziałów RSW - w urzędach pocztowych.

Czytelnicy indywidualni opłacają prenumeratę wyłącznie w urzędach pocztowych i u doręczycieli.

Prenumeratę ze zleceniem wysyłki za granicę przyjmuje RSW "Prasa-Książka-Ruch", Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28, 00-958 Warszawa, konto PKO Nr 1153-201045.

Prenumerata ze zleceniem wysyłki za granicę jest droższa od prenumeraty krajowej o 50% dla zleceńców indywidualnych i o 100% dla zlecających instytucji i zakładów pracy.