JPRS L/9546

12 February 1981

USSR Report

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 6/81)

FBIS FOREIGN BROADCAST INFORMATION SERVICE

NOTE

JPRS publications contain information primarily from foreign newspapers, periodicals and books, but also from news agency transmissions and broadcasts. Materials from foreign-language sources are translated; those from English-language sources are transcribed or reprinted, with the original phrasing and other characteristics retained.

Headlines, editorial reports, and material enclosed in brackets [] are supplied by JPRS. Processing indicators such as [Text] or [Excerpt] in the first line of each item, or following the last line of a brief, indicate how the original information was processed. Where no processing indicator is given, the information was summarized or extracted.

Unfamiliar names rendered phonetically or transliterated are enclosed in parentheses. Words or names preceded by a question mark and enclosed in parentheses were not clear in the original but have been supplied as appropriate in context. Other unattributed parenthetical notes within the body of an item originate with the source. Times within items are as given by source.

The contents of this publication in no way represent the policies, views or attitudes of the U.S. Government.

JPRS L/9546

12 February 1981

USSR REPORT

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 6/81)

CONTENTS

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

HARDWARE

UDC 681.327.537.312.62

MEMORIES AND CRYOELECTRONIC COMPUTER COMPONENTS

Kiev ZAPOMINAYUSHCHIYE USTROYSTVA I KRIOELEKTRONNYYE KOMPONENTY EVM in Russian 1979 pp 2, 93-94

[Annotation and table of contents from the book edited by Doctor of Engineering Sciences G.A. Mikhaylov, Ukrainian SSR Academy of Sciences, Order of Lenin Institute of Cybernetics, Scientific Council on the Problem of "Cybernetics", 320 copies, 99 pages]

[Text] Questions of the refinement of memory devices, their assemblies and components are treated. Considerable attention is devoted to problems of utilizing memories in the design of processors and special purpose systems. The results of theoretical and experimental research in Josephson junction devices in computer and measurement equipment are given, including methods for design calculations and modelling, fabrication technology and research techniques. The collection is intended for specialists engaged in the development of computer equipment.

Table of Contents

[65-8225]

8225
CSO: 1863

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

UDC 681.322.06:517.9

ON ONE METHOD OF ANALYZING THE PRECISION OF SPECIALIZED COMPUTERS FOR DIGITAL SIGNAL PROCESSING

Kiev ZAPOMINAYUSHCHIYE USTROYSTVA I KRIOELEKTRONNYYE KOMPONENTY EVM in Russian 1979 pp 45-48

[Excerpts from the article by V.M. Rovenskiy in the book edited by Doctor of the Engineering Sciences G.A. Mikhaylov, Ukrainian SSR Academy of Sciences, Scientific Council on the Problem of "Cybernetics", Order of Lenin Institute of Cybernetics: "Memories and Cryoelectronic Computer Components"]

[Excerpt] When designing the hardware for specialized computers (SVU) for digital signal processing, it is necessary to take into account effects related to the representation of the coefficients and the numbers being processes with a limited accuracy. This paper proposes the utilization of the method proposed by Welch [1] to analyze the precision of a specialized computer with a fixed decimal point. One of the major merits of the method is its simplicity. We will note that this method was employed by Welch in a special case.

The results of research performed at the Institute of Cybernetics of the UkSSR Academy of Sciences are given in the paper. The proposed method was used to analyze the precision of digital filters with a finite pulse response as well as digital spectrum analyzers operating on the basis of a BPF [fast Fourier transform] algorithm [2]. The selection of these devices specifically for the analysis was due to their widescale application in systems for the automation of scientific investigations.

The examples cited here are a good illustration of the effectiveness of the proposed method in the analysis of the precision of specialized computers for digital signal processing.

BIBLIOGRAPHY

1. Welch, P.D., "A Fixed-Point FFT Analysis", IEEE Trans. on Audio, 1969, V. AU-17, No 6, pp 153-157.

2. Rabiner, L., Gould, B., "Teoriya i primeneniye tsifrovoy obrabotki signalov" ["The Theory and Application of Digital Signal Processing"], Moscow, Mir Publishers, 1978, 848 pp.

3. Oppenheim, A.V., Weinstein, C.J., "Effect of Finite Register Length in Digital Filtering and the FFT", PROC. of the IEEE, 1972, V. 60, No 8, pp 957-976.

[65-8225]

8225
CSO: 1863

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

ELEKTRONIKA-60 LINK TO VIDEOTON-340

Tomsk TEKHNICHESKIYE SREDSTVA VYCHISLETEL'NOY SETI INSTITUTA OPTIKI ATMOSFERY in Russian 1979 pp 69-73

[Article by V. I. Zhuravlev and V. D. Shelevoy: "The Videoton-340 Display in the Communications Line with the Elektronika-60 Computer"]

[Text] Introduction

The Konsul-260 electric typewriter is used as a terminal console in the peripheral equipment of the Elektronika-60 computer. This article proposes a way to connect a Videoton-340 display to the computer channel also. The combined use of these two terminals makes it possible to compensate for the shortcomings that these devices have, for example, to insure a high rate of exchange by means of the display and obtain a document on the electric typewriter. In addition, in the version being proposed the Videoton-340 and electric typewriter have the same external unit address and therefore standard software is fully applicable to operate the display. The selection of the particular unit is done by switching modes on the display console. It should be observed that this scheme offers any of three configurations: computer-typewriter; computer-Videoton-340; computer-typewriter-Videoton-340.

Work [1] considered a similar version of the use of a display with a computer, but it has a number of significant shortcomings, specifically: low rate of information output to the display screen; lack of data input mode from the memory of the Videoton-340 to the computer (the "Send" mode); periodic loss of sign when transfering a line because of lack of correspondence between the time diagrams of the Videoton-340 and the computer, as well as some others. The version we are describing does not have these shortcomings. The working modes are controlled by working mode switch keys on the Videoton-340: "Off Line" — computer on line with the electric typewriter; "On Line" — computer on line with the Videoton-340; "Send" — reading data from the display memory to the computer; "UL" — stop printing. The "UL" key can be used here to stop the exchange of information between the computer and the Videoton-340. This is convenient when outputting large volumes of data to the screen of the Videoton-340 or for a technical shutdown, for example to adjust the punched tape and the like. This mode can also be accomplished without using the "UL" key, but this requires installation of a separate key, for example on the computer control console.

The circuit is built on a series 155 IMS [possibly integrated magnetic circuit]. In terms of design the IMS and communications cable socket are located on the panel of the VI computer control unit.

Description of the Coupling Circuit

In our description of the circuit we will use the terminology adopted in documents [2, 3] without detailed comments. Before describing the coupling circuit let us briefly explain the work of the VI control unit. The part that interests us contains circuits for controlling printing and controlling the keyboard of the electric typewriter.

A. The printer control circuit is designed to receive the codes of signs from the central processor and control the printing mechanism of the typewriter. The sign code goes from the central processor through the computer channel to the input of the data printing register. The code of the sign is entered in the data printing register by the signal "Output 66B" coming from the control signal decoder. From the output of this register the code of signs is transmitted through the decoder to the keys of the lines and columns. The back end of the signal "Output 66B" triggers the printing time cycle shaper. The rate of data output is determined by the speed of the electric typewriter.

B. The keyboard control circuit is designed to receive the code of signs from the combiner of the typewriter, convert the codes of the registers, and transmit the code to the central processor. The signal pulse SiI launches a circuit that forms a strobing pulse which authorizes the use of permanent storage and enters the code in the keyboard data register; at this time the TRKL V [expansion unknown; "KL" = keyboard] signal is formed. In response to this signal the typewriter issues the signal "Input 62V," which opens the output gates of the keyboard data register, the signal "Input (52 + 62) N" switches on the transmitter, and the code of the symbol proceeds to the central processor.

C. Figure 1 below shows the diagram for data output to the Videoton-340 unit. The diagram shows fragments of the VI control unit circuit whose elements are numbered, for example RG-D17. The elements that are not numbered are part of the coupling circuit being described.

The code of the sign is taken from the output of keyboard data register D17 and fed through a cable up to two meters long to the coupling panel of the Videoton-340 in the "On Line" mode. In this mode the time cycle shaper of typewriter use and the permanent memory output units D13, D15, and D16 are blocked. The time diagram of computer work with the Videoton-340 is determined by the signals "DMDI" and "STRBI." Shaper "S" is used to adjust the time diagram. It issues the signal "Request 66V" to the Videoton-340 for the delay time of signal "BMDI."

D. Figure 2 below shows the circuit for feeding data from the display to the computer. The sign code from the output register of the Videoton-340 passes through the "OR wire" circuit formed by IMS 155 LA8 on the coupling panel and the input units of permanent memory, to the input of keyboard data

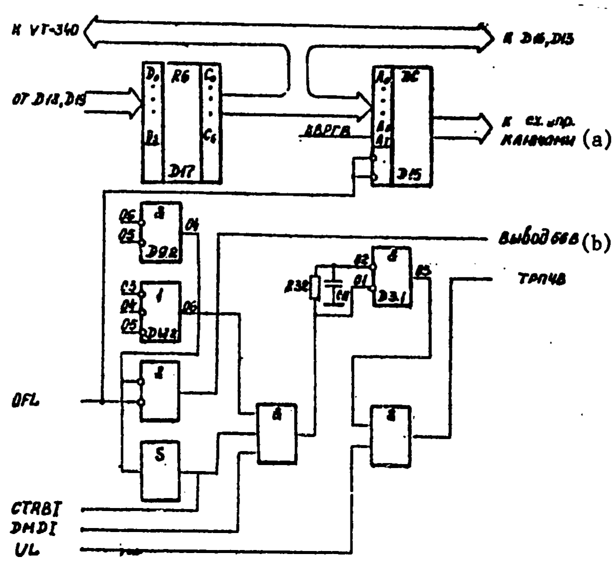


Figure 1

Key: (a) To Key Control Circuit;
(b) Output 56V.

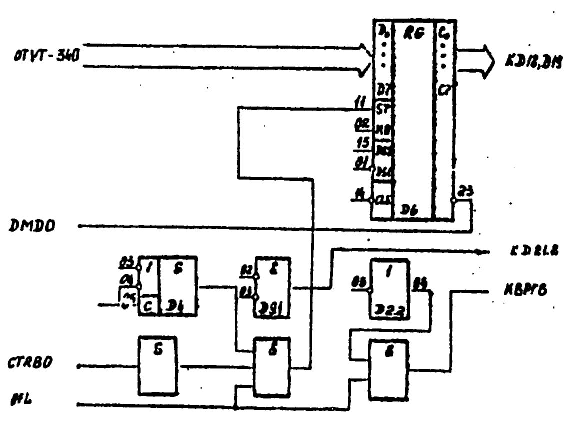


Figure 2

register D6 and is recorded in it on signal "STRBO." The inputs of permanent memory and the mode switch are blocked in this case. The signal "DMDO" is taken from the output of RG-D6 and confirms receipt of the code of the keyboard data register. At the signal "Input 52V" the code of the sign passes through the line shapers to the central processor.

E. To establish the "Stop Printing" mode contact No 4 MS15 on the VD-06 panel must be moved to plug A07 (socket SN8) and connected with contact V38 (socket SN10). The computer and the display are connected by 21 paired cables.

F. Conclusion

This article has described the coupling of the Videoton-340 display with a series-produced Elektronika-60 computer. The modification done on the panel of the VI control unit has absolutely no effect on the work of the computer in the absence of one or two terminals. The simplicity of the circuit and small number of elements used make it possible for any user of this computer to accomplish this coupling easily.

The authors express their gratitude to Marina Nikolayevna Vykhodsteva and Vladimir Ivanovich Pelipenko for detailed discussion of the question investigated in this article.

BIBLIOGRAPHY

1. Vykhodtseva, M. N., and Pelipenko, V. I., "Use of the Videoton-340 Display as an Operator's Console for the Elektronika-60 Microcomputer," Preprint FIAN, 1979.

2. "The Elektronika-60-15VB-6 Electronic Computing Machine. Technical Description," Production Research Department of the Eletronika Central Scientific Research Institute, 1977.

3. "The Videoton-340 Type Display. Technical Description," Computer Technology Plant, Budapest, 1978.

[61-11176]

11,176
CSO: 1863

UDC 681.39:327.8

CONSTRUCTION OF A COMMUNICATIONS PROCESSOR FOR A MODELING COMPLEX

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 25-28

[Article by O. A. Kondakov, L. N. Matveyeva, O. M. Omarov, T. E. Temirkhanov, and V. M. Khachumov, of Dagestan Polytechnic Institute, Makhachkala]

[Excerpts] The problem of information communication between man and computer, between various computers of a system or between computers and peripherals of various kinds is acquiring paramount importance. This has a substantial influence on the effectiveness of productivity of system processors.

One possible approach to solution of the problem is the creation of communications processors which assure performance of the following tasks [1.2]:

--local processing and storage of information;
--creation of the possibility of flexible and economical coupling with data input devices;
--bringing computer capacities closer to the users;
--reduction of the currently high cost of communications through the concentration of data.

Presented below is one of the possible solutions of the question of communication processor construction, one recommended by the authors for introduction into an analog-digital computer complex consisting of a number of digital and analog computers and intended for simulating complex dynamic objects in real time.

Communication processor technical data

Digit capacity of data processing track and exchange in communications module	16 + 1 control
Digit capacity of short instructions	16 bits
Digit capacity of long instructions	32 bits
Place of arrangement of instructions (the instructions list includes instructions of operations between registers, instructions for the control and switching of states and instructions for transitions)	main memory field
Number of general-purpose registers	16
Number of special-purpose registers	16
Maximum number of switched communication modules	8

Number of directions in communication modules, direct and reverse	up to 64
Number of subscribers connected to direct access channels	up to 256
Addressable main memory field	64K 16-digit words
Main memory field capacity	4K 16-digit words

The communications processor is made with K133 integrated microcircuits.

[54-2174]

2174
CSO: 1863

PLOTTER AND DISPLAY SPECIFICATIONS

Moscow IZMERENIYA, KONTROL', AVTOMATIZATSIYA in Russian No 9-10, 1980 pp 54-61

[Tables from the article by Candidate of the Engineering Sciences V.P. Mikheyev and Engineer P.I. Savostin: "Graphical Information Documentation Devices"]

[Excerpt] Table 1. Characteristics of Electromechanical Plotters

Model, Company, Nation	Size of the Working Field, mm	Unit Step, μm	Precision (Reproduceability) ± μm	Drawing Speed m/min
	Plane Table Plotters			
8446, 8448, Aristo, FRG	From 1200 x 1200 to 3000 x 3000	30	150	From 2.5 to 4
Coradomat 21, Coradi AG, Switzerland	1300 x 1600	40	100	-
Z-92, Zuse, FRG	From 1200 x 1400 to 2000 x 7800	50	100	6.6-13.2
500, 7000, Calcomp, U.S.	From 700 x 860 to 1200 x 1830	12.5-250	150	300-2600 steps per sec
1200, Xynetics, U.S.	1450 x 2260	25	130	60
32, 33, Gerber Scientific Instruments, U.S.	1220 x 1220 610 x 610	-	23; 12	From 1.5 to 5
YeS 7051, USSR	1090 x 1200	25	150	0.5
EM703, USSR	1200 x 1200	-	50	5.5
EM712, USSR	700 x 1200	100	100	10
	Roller Plotters			
121, Benson, France	740 x 100,000	100	100	5.4
1934/6, Calcomp, U.S.	735 x 36,000	12.5	-	6
YeS7032, USSR	420 x 80,000	50; 100	200	2
YeM711, USSR	420 x 40,000	100	100	10

Table 1. [Continued]:

Model, Company, Nation	Size of the Working Field, mm	Unit Step, μm	Precision (Reproduceability) ± μm	Drawing Speed m/min
	Photoplotters (Image Generators)			
745, 728, California Computer Products, U.S.	1100 x 1500 500 x 500	12.5 12.5	25 (10.0) 30	-
Superior Electric, U.S.	300 x 450	2.5	0.25 (0.12)	-
LSI 610, Optomechanisms, U.S.	100 x 100	-	0.5 (0.25)	-
EM549, USSR	140 x 140	-	0.5 (0.3)	-
EM519B, USSR	80 x 80	-	2.0 (1.0)	-

Table 2. Characteristics of CRT Graphic Displays

Model, Company, Nation	Film Width (Size of the Working Field) mm	Resolving Power lines/mm	Speed
YeS-7602, Robotron, GDR	16	35	10^5 char/sec
MAS-90, Magen, Great Britain	16, 35	80	75 frames/min
300, Benson, France	16, 35	6,700 lines/raster	$2 \cdot 10^5$ points/sec
"Foton", USSR	(24 x 18)	80	30 μsec
"Karat", USSR	(24 x 18)	Up to 100	10^5 points/sec

[58-8225]

8225
CSO: 1863

VK-1010 RELIABILITY

Moscow VYCHISLITEL'NYYE SISTEMY in Russian 1980 p 191

[Excerpt from book by Ivan Vasil'yevich Panfilov and Anatoliy Mikhaylovich Polovko, Izdatel'stvo "Sovetskoye Radio", 8000 copies, 304 pages]

[Excerpt] It is necessary to compute the quantitative characteristics of the reliability, readiness and technical efficiency of the computing complex a diagram for the calculation of whose reliability is presented in fig 11, where the following symbols are used: 1 and 2 represent the processor, magnetic tape storage units and input/output units of the first and second computers, respectively; 3 and 4 represent the external magnetic disk storage; and 5 and 6 represent the external magnetic tape storage. With certain assumptions it is possible to reduce to this diagram the functioning of the VK-1010 computing complex, constructed on the basis of two type YeS-1030 digital computers (a description of this complex is given in the book "Elektronnaya vychislitel'naya mashina YeS-1030" [The YeS-1030 Computer] edited by A.M. Larionov, Moscow, Statistika, 1977).

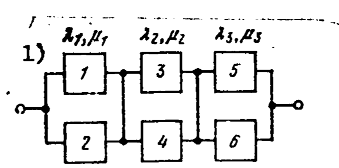


Figure 7.11. Structural Diagram of Computing Complex

Key:

1. λ = failure rate and μ = restoration rate

Values of the failure and restoration rates of calculation elements are summarized in table 7.1.

Table 7.1.

Number of calculation element	1 and 2	3 and 4	5 and 6
λ , h^{-1}	$8 \cdot 10^{-3}$	$4.6 \cdot 10^{-3}$	$2.2 \cdot 10^{-3}$
μ , h^{-1}	1	1	1

[77-8831]

8831
CSO: 1863

UDC 681.326.7

INTERFACING VIDEOTON-340 DISPLAYS TO THE BESM-6 COMPUTER AT A TRANSMISSION RATE OF 1,200 BAUDS

Moscow ALGORITMICHESKIYE MODELI V AVTOMATIZATSII ISSLEDOVANIY in Russian 1980 signed to press 27 Jun 80 pp 235-240

[Article by G.A. Serdukov in the book edited by Doctor of the Engineering Sciences Professor V.M. Ponomarev, Izdatel'stvo "Nauka", 3,950 copies, 255 pages]

[Text] The design of collective use systems which provide for user access to the computer capacities through terminals makes it possible to increase the efficiency of users' working with algorithmic models.

A Videoton-340 (Hungary) display can be used as a terminal for the BESM-6 computer; this display has high reliability, low noise in operation, a high data exchange rate with the computer (up to 1,200 bauds) and considerable capabilities for data editing. The interfacing of the Videoton-340 with such an interchange rate to the BESM-6 computer makes it possible to maximally utilize the terminal. At the present time, a variant for interfacing via a telegraph channel using external synchronization has become rather widespread. It should be noted that this method cannot called the best one, since the Videoton-340 display was designed for asynchronous operation. However, the comparatively simple circuit design using external synchronization, which does not entail considerable reworking in the BESM-6 computer and which can be accomplished by any organization, undoubtedly has an advantage over other interface variants.

The article published in the collection "New Hardware for the BESM-6"* was taken as the basis of the design for the circuit to interface the Videoton-340 to the BESM-6 at a speed of 1,200 bauds. The basis for the design is the principle of

*Lopatnikova T.M., Mikhaylov G.M., Sychkov Yu.A., "Interfacing Videoton-340 Type Displays to the BESM-6 Computer", in the book "Novyye Sredstva Apparatnogo Obespecheniya BESM-6" ["New Hardware for the BESM-6"], Moscow, Computer Center of the USSR Academy of Sciences, 1976.

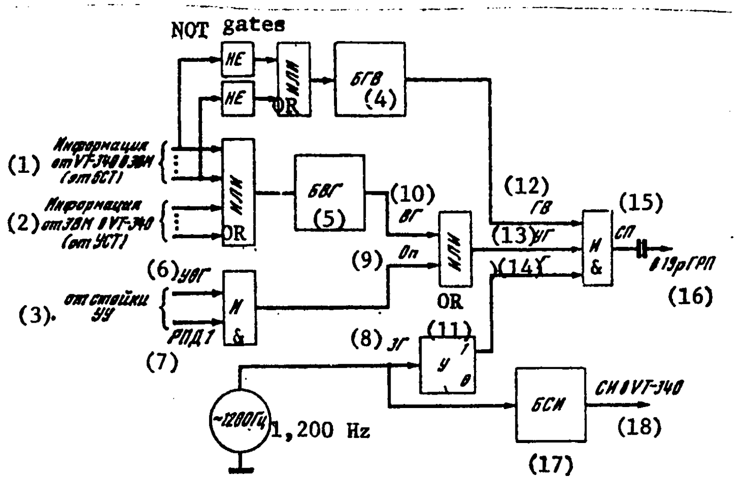


Figure 1.

Key:

1. Data from the VT-340 to the computer (from the BST) [not further defined];
2. Data from the computer to the VT-340 (from the UST) [not further defined];
3. From the UU rack [not further defined];
4. BGV [Videoton readiness unit];
5. BVG [unit for turning on the generator];
6. UOG [not further defined];
7. RPD1 [not further defined];
8. ZG [?master oscillator?];
9. Query signal;
10. Generator on signal;
11. U [not further defined];
12. Videoton readiness signal;
13. UG [not further defined];
14. G [not further defined];
15. SP [?interrupt signal?];
16. To the 19th digit of the GRT [main interrup register];
17. Pulse synchronization unit;
18. Sync pulses to the VT-340.

* * *

common synchronous operation of all channels with the output of codes from the Videoton, as a result of which, data can be received in the BESM-6 simultaneously from all displays connected to the telegraph channels. The synchronizing signals are generated in this case by an external generator, which is synchronized with the central processor of the BESM-6 through a system of interrupts. The 19th digit is singled out in the main interrupt register (GRP) for operation with the channels of the Videotons. The principle of synchronous operation imposes the condition of external forced synchronization on the operation of the telegraph interface of each Videoton in the code output mode, while the code receive circuit from the channels is designed in each device taking into account the asynchronous operation of the channel.

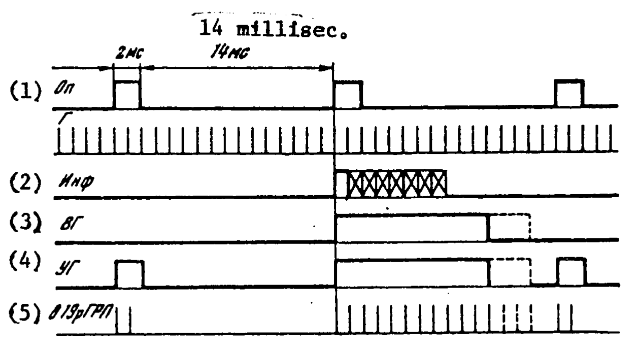


Figure 2.

Key: 1. Query signal;
2. Data signal;
3. VG [?Videoton readiness?];
4. UG [not further defined];
5. To the 19th digit of the main interrupt register.

When the Videoton is interfaced using this variant, the data transmission speed when working with the Videoton is governed by the frequency of the external generator and has been chosen as 250 bauds.

As a result of the tests which have been conducted in the LNIVTs of the USSR Academy of Sciences, this circuit remains operable at a transmit rate of 1,200 bauds also. However, when operating in the on-line mode, when the interrupts are fed to the 19th digit of the GRP at a frequency of 1,200 Hz, regardless of whether there is data for interchange between the Videoton and the BESM-6 or not, the computer goes over to the data exchange program between the computer and the telegraph channels at a frequency of 1,200 Hz. As a result, the processor time losses amount to 15 percent.

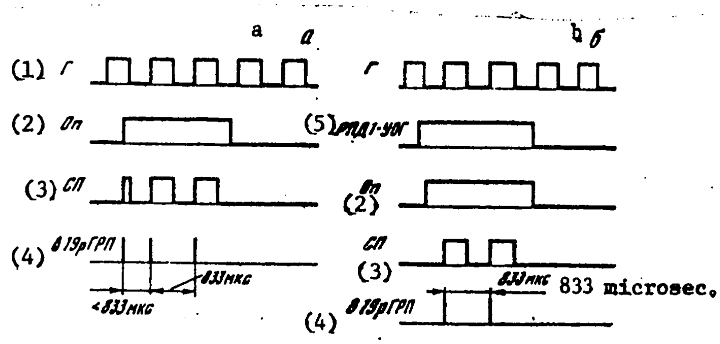


Figure 3.

Key: 1. G [clock pulses];
2. Query signal;
3. SP [?interrupt signal?];
4. To the 19th digit of the GRP [main interrupt register];
5. RPD1-UOG [not further defined];

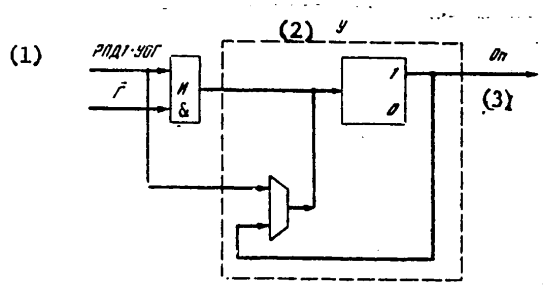


Figure 4.

Key: 1. RPD1-UOG [not further defined];
2. U [?amplifier?];
3. Query signal.

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

The goal of this work was to design a circuit which would feed interrupt signals to the 19th digit of the GRP [main interrupt register] at a frequency of 1,200 Hz only when exchange information is present between the computer and the Videoton. Tests of the designed circuit showed that the time losses of the central processor were cut down to 1 to 2 percent.

The circuit for interfacing the Videoton to the BESM-6 at a transmit rate of 1,200 bauds is described below.

The logic circuit and the time diagram for the generation of the interrupt signals at the 19th digit of the GRP are shown in Figures 1 and 2.

When exchange data is absent between the computer and the Videoton, the "Query" (Op) signal is generated at a frequency of 62.5 Hz with a width of 2 milliseconds. The query signal is formed by means of the UOG and RPD1 signals, where these units are incorporated in the counter of the frequency divider (the UU rack) [expansions not further defined]. It can be seen from Figure 1 that when the query and "Videoton Ready" (GV) signals are present- 2 or 3 interrupt signals (SP) are fed to the 19th digit of the GRP (the period of the interrupt signal is 830 microseconds). The Videoton readiness signal is fed when power is applied to the Videoton and the computer, interrupt signals are fed to the machine at a frequency of approximately 150 Hz, something which accounts for the time losses of the processor of 1 to 2 percent which were mentioned above.

When exchange data is present between the computer and the Videoton, during the processing of the first interrupt signal called up by the query signal, a start signal is generated for the data which is fed to the input of the unit which turns on the generator (BVG) (see Figure 1). When the start signal is fed to the input of the BVG, a "generator on" signal is generated at the output, with a width of no less than 10 milliseconds. This signal enables interrupt signals at a frequency of 1,200 Hz to continue for the exchange of no less than one symbol of data. Since the start signal has the polarity of a logic "zero" of the Videoton code, then depending on the symbol code, the width of the generator-on signals can be increased.

Thus, the interrupt siganls at the 19th digit of GRP are fed at a frequency of 1,200 Hz only when exchange data is present between the computer and the Videoton.

The following difficulty came up during the realization of the circuit described here. If the query signal arrives at the point in time when the "generator" (G) signals is positive, then the first interrupt signal is incorrectly generated. It can be seen from Figure 3a that the amount of time between the first and second interrupt signals will be less than 833 microseconds, as a result of which, errors occur during information exchange between the computer and the Videoton. The following logic circuit was designed to eliminate this phenomenon (Figure 4). In accordance with this circuit, the query signal can be generated only when the generator signal has a negative polarity, and as can be seen from Figure 3b, the duration between the first and second interrupt signals is always equal to 833 microseconds.

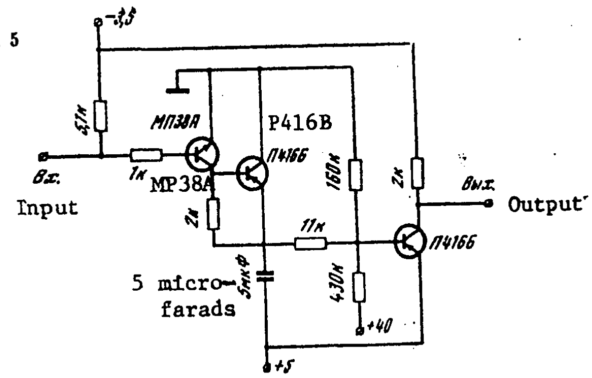


Figure 5.

The logic circuit shown here was based on the system used in the BESM-6: "diode logic--paraphase output amplifier".

A description of the unit for turning on the generator (BVG) and the Videoton readiness unit (BGV) is given below.

The circuit of the sync pulse generator (BSI) was taken from the article indicated above without any changes.

The Unit for Turning On the Generator (BVG)

The basic schematic of the BVG is shown in Figure 5. When there is no data present at the input to the MP38A transistor, a negative signal is produced, as a result of which, all transistors are cut off. The voltage across the plates of the capacitor is close to zero. When a start signal (positive polarity) arrives from the BST and UST circuits [not further defined], all transistors turn on, a positive potential appears at the output and the capacitor is simultaneously charged up to +5 volts. Following the completion of the start signal, the P416B output transistor remains turned on during the discharge time of the capacitor. The time the output transistor remains on

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

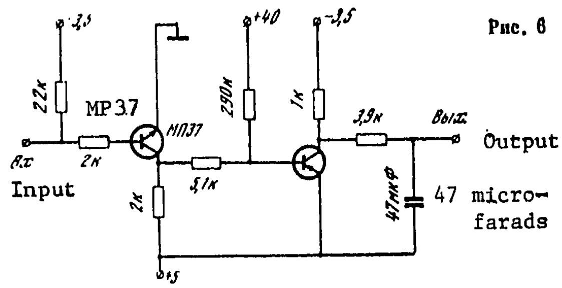


Figure 6.

after the completion of the start signal should be no less than 10 milliseconds, something which is sufficient for the exchange of one symbol between the computer and the Videoton.

Thus, during continuous data exhange (symbol by symbol), the capacitor will be charged up to +5 volts all the time, and the output transistor will thereby be turned on for the entire time of data transmission.

The Videoton Readiness Unit (BGV)

The basic schematic of the BGV is shown in Figure 6. Inverted signals are fed from the outputs of the BST circuits (the circuits for data transmission from the Videotons to the BESM-6) are fed to the input of the BGV. When the power is turned on even if for just one Videoton, a positive potential will appear at the input to the BGV. The MP37 and P416B transistors are turned on in this case, and a positive potential will appear at the output of the BGV, something which enables the generation of interrupt signals at the 19th digit of the GRP. The size of the capacitance was chosen so that it does not allow a negative potential to appear at the output of the BGV during data transmission. When the power is cut off for all Videotons, a negative potential is fed to the input of the BGV and transistors MP37 and P416B cut off. In step with the charging of the capacitor, a negative potential appears at the output of the BGV, something which prohibits the generation of interrupt signals at the 19th digit of the GRP.

[64-8225]

8225
CSO: 1863

UDC 681.326:519.95

MICROPROCESSOR CONTROL SYSTEMS FOR MANIPULATION ROBOTS BASED ON THE CAMAC STANDARD

Moscow ALGORITMICHESKIYE MODELI V AVTOMATIZATSII ISSLEDOVANIY in Russian 1980
signed to press 27 Jun 80 p 253

[Abstract of article by Domoratskiy, A.N., Kulakov, F.M. and Surma, S.V.]

[Text] Questions of the design of robot systems using microprocessors based on the CAMAC standard are treated. An example is given of the realization of an efficient control level for manipulation robots based on a CAMAC crate, and an algorithm for the generation of the control action for drives which initiate the motor activity of the robot is analyzed, taking into account the specific features of the CAMAC interface. Data are given on foreign microprocessors made in the form of standard CAMAC modules. The possibility of the CAMAC based design of microprocessor control systems with a distributed architecture is described. Tables 1; figures 5; 3 bibliographic citations.

[64-8225]

8225
CSO: 1863

RUMANIAN MINICOMPUTERS ADVERTISED

Riga AVTOMATIKA I VYCHISLITEL'NAYA TEKHNIKA in Russian No 6, 1980 p 100

[Advertisement from Vneshtorgreklama [Foreign Trade Advertising] All Union Association]

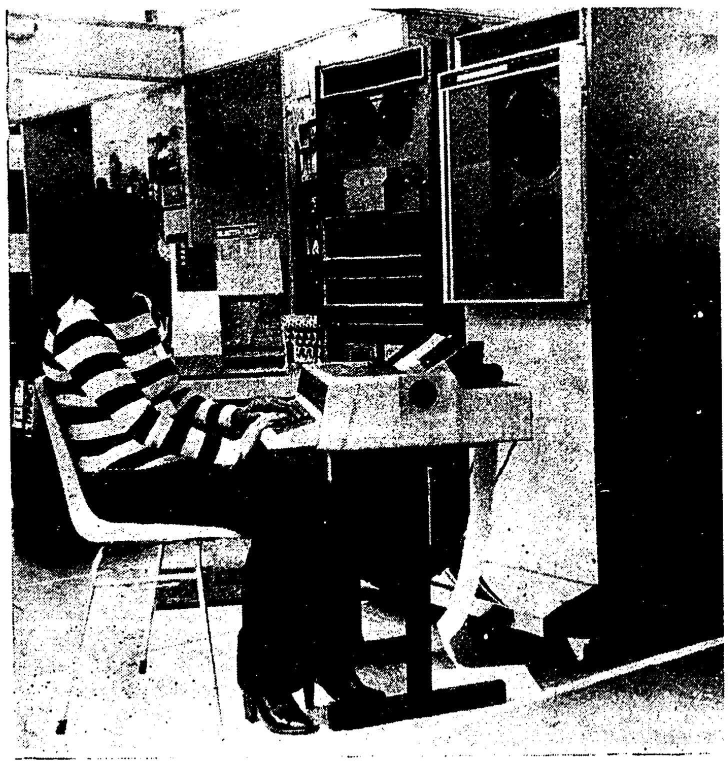
[Text] Need to solve your problems by means of minicomputers? We offer you an ideal solution: Our "Independent-100," "Coral 4001/4011" and "M-18" are at your disposal!

Because of logical diversity in the use of peripheral equipment these new minicomputers, which have a ferrite core storage and an RAM (I-100), as well as an RAM, PROM and REPROM (M-18), can be used in different fields of work, such as in controlling technological processes, in scientific research and in technical and scientific calculations, in processing commercial data, in data transmission, in teleprocessing data in an on-line and off-line connection with FELIX C-256/512 systems at information concentration stations, in a computer-coordinated training process, etc.

For additional information and detailed documentation contact the exporting firm:

Electronum Import and Export, specialized foreign trade enterprise, 70317 Bucharest, Rumania, 11-15 Olari Street, P.O. Box 34-36; Telex 11547, 11584.

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9



[70-8831]

8831
CSO: 1863

UDC 62-181.4

CONSTRUCTION OF HIERARCHIC STORAGE SYSTEMS FOR MICROCOMPUTERS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 16-21

[Article by Yu. S. Yakovlev, candidate of technical sciences, Institute of Cybernetics, UkSSR Academy of Sciences, Kiev]

[Text] The steadily growing demand for microcomputers, the sharp expansion of the area of their application and the complication and increase of volume of the problems being solved by microcomputers naturally lead to increase of the required storage capacity. The requirement for data processing speed also increases. However, the requirements of low cost and high reliability remain unchanged for microcomputers; in most cases this stipulates their application in systems with various purposes. The microcomputer storage in that case is the main determining factor in satisfying the above-indicated requirements, since in the number of crystals (BIS--large-scale integrated microcircuits), dimensions, weight, power consumption and other parameters it considerably surpasses a processor which can be realized on a single crystal at the present level of development of integrated technology.

One of the central problems in storage construction is that of realizing a storage with large capacity, high productivity and low cost. One known way to solve this problem which has been confirmed by experience in the development and application of digital computers of various classes is the creation of hierarchic storage systems.

A hierarchic storage system is a complex system with a large number of variables which generally is not amenable to precise mathematical description.

There exists a large number of various theoretical models of hierarchic systems with different simplifications and assumptions assuring qualitative rather than quantitative evaluations of given architectural and structural decisions: a model based on queueing theory [1], a model of cyclic alternation [2], linear models [3,4], etc [5-8].

Theoretical investigations of those models have been scattered in various literature sources and have been burdened with complex theoretical computations, and as a result of that are unsuitable for practical application. Also lacking is an engineering procedure for calculating the main parameters of multilevel hierarchic storage systems. The procedure presented in [7] is suitable for estimating the parameters of only two-level storage systems.

In the present work an attempt is made to present the results of analysis of some theoretical models in a form convenient for practical use in the development of microcomputers or controllers based on them. In that case the main attention has been given to so-called sequential models of a hierarchy as the most accessible for use and at the same time sufficiently reflecting the essence of the processes occurring in a hierarchic system. A procedure is proposed for calculation of the main parameters of a hierarchic system.

The sequential model [3] contains n levels $M_1, M_2, \ldots, M_n$, connected in series. Information is transmitted between adjacent levels. In that case:

--if the required page is in M_i, there is a copy of it in each of the lower levels $M_{I+1}, \ldots, M_n$;
--if the required page is not in M_i, an inquiry is sent to all sequentially lower-lying levels until it is found in some one of them;
--if the level M_i is filled and it is necessary to introduce a new page into it from M_{i+1}, then one of the pages in M_i is erased and in its place is sent a page from level M_{i+1} (in accordance with the adopted algorithm of displacement).

The maximum amount of information that can be stored in the system is equal to the storage capacity of the lowest level.

The following assumptions have been adopted for that model: the strategy of storage control is determined by the success function* (the entry coefficient) H(C); the technology of the device is characterized by the access time of the device and the cost per unit of storage (byte or other unit); the success function H(C) and the cost function b(t) of the device are represented correspondingly as a function of capacity and a function of the speed in the following form:

$$H(C) = 1 - F(C), \quad F(C) = F_0 C^{-\alpha}, \quad b(t) = b_0 t^{-\beta},$$

where F_0 is a unit of capacity; b_0 is a unit of cost; α and β are the steepness of function of the loss coefficient F(C) and the cost function b(t) respectively.

Such a representation is legitimate, since according to the results of research [3,4] it agrees well with practical data for a broad class of algorithms and types of storage devices.

The task of optimization. Let the optimization criterion be minimization of the effective hierarchy access time T under the condition of limitations on cost S_0 and capacity C_n of the storage system. In that case the task of optimization is formulated as the task of nonlinear programming with minimization of the function

$$T = t_1 + \sum_{i=2}^{n} F(C_{i-1}) t_i = \sum_{i=1}^{n} F(C_{i-1}) t_i \qquad (1)$$

under the following limitations:

$$S = \sum_{i=1}^{n} b(t_i) C_i < S_0. \qquad (2)$$

*The success function H(C) is the probability of finding the required information in the storage capacity C.

$t_i > 0$ for $i = 1, 2, \ldots, n$; $C_i > 0$ for $i = 1, 2, \ldots, n-1$, where n is the number of levels of the hierarchic storage system; $F(C_{i-1})$ is the loss function (probability of defect) during reference to the (i-1)-th level; $t_1, t_2, \ldots, t_i, \ldots, t_n$ are the access times of the corresponding hierarchy levels; S_0 is the cost of the storage system; C_i and C_{i-1} are the storage capacities of the i-th and (i-1)-th levels; $b(t_i)$ is the function of the technology cost of the i-th level.

In [3] it was shown that without violation of generality it can be assumed that $F_0 = 1$ and $b_0 = 1$. Then expressions (1) and (2) assume the forms:

$$T = t_1 + \sum_{i=2}^{n} C_{i-1}^{-\alpha} t_i; \quad (3) \qquad S/S_0 = \frac{1}{S_0} \sum_{i=1}^{n} t_i^{-\beta} C_i < 1, \quad (4)$$

It is characteristic of multilevel storage systems of microcomputers that the storage of the first level can be arranged either on a single plate or directly on a single crystal with a processor [9]. In the latter case the first-level storage is made according to the same technology as the processor and correspondingly has a completely determined speed. When arranged on a single plate with a processor the storage speed can also be considered known, as it is determined mainly by optimum matching of the time charts of functioning of the system storage-processor. In that case the solution of the task of optimization with respect to the sequential model of a hierarchic storage system is simplefied somewhat, as the value of the parameter t_1 has been predetermined, and on the value of the parameter of the storage capacity C_1 a limitation has been imposed from above, one stipulated by the allowable degree of integration (the level of technology of large-scale integrated microcircuit production) and design-technological limitations of printed circuit production.

With respect to the more general case it is advisable to solve the task of optimization when the time t_1 is given and it is necessary to determine the following: the value of the first-level storage capacity $C_1 \leq C_{1\text{-all}}$; the optimum number of storage levels n, the cycle time t_i, the capacity C and the cost b_i of each i-th level of the hierarchy during limitations on the cost S_0 and storage capacity C_n of the system.

Expressions obtained in [4] can be used to determine those parameters.

The optimum number of hierarchy levels n is determined as:

$$n = 1 + \frac{\alpha\beta - 1}{(1 + \beta) \ln \alpha\beta} \ln \left(\frac{C_n}{C_1} \right). \quad (5)$$

Then the first-level storage capacity C_1 is equal to

$$C_1 = \frac{F_0^{1/\alpha} S_n t_1^{\beta} (\alpha\beta)^{n-1} (\alpha\beta - 1)}{b_0 [(\alpha\beta)^n - 1]}. \quad (6)$$

If the values of the parameters t_1 and C_1 for the first level of a hierarchy are known, the values of the parameters t_i and C_i for each i-th level can be determined in the following manner:

$$\ln\left(\frac{C_i}{C_1}\right)=\frac{1-(\alpha\beta)^{i-1}}{1-(\alpha\beta)^{n-1}}\left[\ln\left(\frac{C_n}{C_1}\right)-k(n-1)\right]+k(i-1), \qquad (7)$$

where

$$k=\frac{(1+\beta)\ln\alpha\beta}{\alpha\beta-1}.$$

$$t_i=t_1[(\alpha\beta)^{i-1}C_i/C_1]^{1/\beta}. \qquad (8)$$

The distribution of prices b_i and the time of use d_i of adjacent levels of the storage hierarchy is accomplished in accordance with the expression

$$b_i/b_{i+1}=d_i/d_{i+1}=\alpha\beta. \qquad (9)$$

The optimum access time of a hierarchy can be expressed through the storage capacity C_1 and the access time t_1 of the first level in the following manner:

$$T_{opt}=t_1+t_1(\alpha\beta\sigma_2)^{1/\beta}\cdot C_1^{-\lambda}\cdot C_n^{\sigma_n/\beta}\,T_g(n,\alpha,\beta), \qquad (10)$$

where

$$\lambda=\alpha c_2+\beta^{-1}=\beta^{-1}\frac{(\alpha\beta)^n-1}{(\alpha\beta)^{n-1}-1};$$

$$T_g(n,\alpha,\beta)=(\alpha\beta)^{\eta}/\sigma_n^{1+1/\beta};$$

$$\eta=(1+\beta^{-1})\left[\frac{\alpha\beta}{\alpha\beta-1}-\frac{(n-1)(\alpha\beta)^{n-1}}{(\alpha\beta)^{n-1}-1}\right];$$

$$\sigma_n=\frac{1-\alpha\beta}{1-(\alpha\beta)^{n-1}};\quad \sigma_2=(\alpha\beta)^{n-2}\sigma_n.$$

Thus the above presented formulas for determination of the main parameters of a hierarchic storage system reflect an explicit dependence of those parameters on the steepness α of the loss function $F(C)$, the steepness β of the capacity of the storage system C_n and its total cost S_0.

Determination of the values of the parameters α and β represents an independent task in the general case. In particular, the success (entry) function in the i-th level during access to a reference present on a page does not depend only on the storage capacity of that level but also on the page size and the algorithm for page replacement, that is, in principle on the algorithm for control of the storage system and a number of other factors. But since the success function $H(C)$ is connected with the loss function, then parameter α also depends in the general case on the above-indicated factors.

Several storage control strategies have been developed and are finding application at the present time. According to a model [3], a processor in the time t presents the hierarchic storage system a request for page x_1. If x is in the upper-level storage of the hierarchy M_1, then that request can be satisfied without replacing the page in M_j. If the upper-level storage is filled, x replaces a certain page y in M_1. The selection of page y_i is made in accordance with the adopted replacement algorithm. There is a separate class of replacement algorithms, defined in [10] as a class of stack algorithms.

For the arrangement of any of the pages, provision has been made in the system for corresponding page zones, so that each page at any time can be disposed in only one page zone. A cartographic function is attributed to each hierarchic level and indicates for each page a page zone which it can occupy in the level. Such a function can be unlimited (any page can occupy any page zone), completely limited (any page can occupy only one definite page zone) and partially limited (all other cases).

Determination of parameters α and β. If the algorithm for page replacement and the cartographic function of a hierarchic sysem are known the functions H(C) and F(C) can be determined and, consequently, the value of parameter α. A necessary condition for that is the presence of a series of requests for pages $x=x_1, x_2, \ldots, x_n$ (in accordance with the algorithm for solving the problem), which we will conventionally call the address track. If the digital computer is specialized for the solution of one or several specific problems, such a sequence is obvious and is very simply compiled. However, if the digital computer is specialized for the solution of a class of problems, then that address track is determined for a standard task. One way to obtain the success function for a given address track is simulation [10], provided the first-level storage capacity

$$C_{page} \leqslant C_1 \leqslant C_{1-all},$$

where C_{page} is the page size (in bytes).

The coefficient of technology cost β can be determined by approximating the function $b(t) = b_0 t^{-\beta}$, constructed on the available data of developments of various types of storage. The same function can also be constructed however, for storage modules of a microprocessor set on the basis of their parameters of storage capacity, cost and speed.

Procedure for calculating the main parameters of a hierarchic storage system. 1. The values of parameters α, F_0, β and b_0 are determined. In that case the parameters α and F_0 are determined by simulation for a given address track, reflecting either an algorithm for solving a specific problem (for a specialized digital computer) or an algorithm for solving a typical problem (for a digital computer orientated toward solving a definite class of problems). In the simulation the values of the first-level storage capacity $C_{page} < C_1 < C_{all}$ are taken, and one of the algorithms for page replacement (for example, in the first stage the storage system equation simplest from the point of view of practical realization of the device can be taken) [10].

The dependence of the value of the error function F(C) on the value of the storage capacity is constructed* and a corresponding analytical expression is found for it.

*The functions F(C) and b(t) must have the form $F(C) = F_0 C^{-\alpha}$; $b(t) = b_0 t^{-\beta}$. Only in that case can the above-presented expressions be used.

The dependence of the cost per unit of information $b(t)$ on the speed for various types of available developments of storage modules is constructed: a corresponding analytical expression is found for it and the values of parameters β and b_0 are determined.

2. The values of the initial parameters of the storage capacity C_n and cost S_0 lead to a normalized form.

3. With it taken into account that the value of the parameter t_1 was determined with consideration of requirements on the part of the processor, the storage capacity C_1 is calculated with expression (6).

4. The optimum number of hierarchy levels n, the values of the parameters of storage capacity and speed for each level and also the equivalent access time T_{opt} of the hierarchic system are determined with expressions (5), (7), (8) and (10).

If the value of T_{opt} does not assure the required processor productivity, then it is necessary above all to review the algorithm for page replacement (if necessary, also the page size), so that when point 1 of the given procedure has been performed the value of parameter α may be increased. In that case points 2-4 of the procedure are performed again.

5. With $S_0=\sum_{i=1}^{n} b_i$ taken into account, the distributions of cost and time of use are determined for all levels of the hierarchic system with expression (9).

6. On the basis of the obtained parameters C_i, t_i, b_i, $i = 1, 2, \ldots, n$, for each hierarchy level in accordance with the graph of $b(t)$ the corresponding storage modules are determined, on the basis of which a hierarchic storage system with the required technical characteristics must be realized. In that case, naturally, the value of the storage capacity of a specific module $C_{i(M)}$ for its application in the i-th level of the system must be greater than (or equal to) the value of C_i obtained with the cited formulas, that is, $C_{i(M)} \geqslant C_i$, and for the values of parameters of access time $t_{i(M)}$ and cost $b_{i(M)}$ of the selected module the reverse correlations must be fulfilled, that is,

$$t_{i(M)} < t_i, \quad b_{i(M)} < b_i .$$

It can turn out that for any i-th level the set Θ_i of storage modules with the parameters $\{y_j\}$, $j = 1, 2, \ldots, \ell$ correspond to the values $C_{i(M)} \geqslant C_i$, $t_{i(M)} \leqslant t_i$, and $b_{i(M)} \leqslant b_i$. Then a single specific module from the set Θ_i can be selected in accordance with the complex criterion [11]:

$$Z(m_j) = \sqrt{\sum_{j=1}^{l} \left[p_j \frac{y_j - y_j(m_k)}{y_{j(\max)} - y_{j(\min)}} \right]^2},$$

where $y_j(m_k)$ is the value of the j-th parameter of the k-th storage module in the set Θ_i; $y_{j(\max)}$ and $y_{j(\min)}$ are the maximum and minimum values respectively of the j-th parameter of the storage module, determined on the truncated set* of variants $\Theta^{(1)} \subset \theta_i$; y_j is the value of the j-th parameter (the minimum if the parameter is minimized, and the maximum if the parameter is maximized) determined on the set of storage modules θ_i; p_j are the coefficients of importance of the j-th parameter (they can be determined in accordance with the procedure described in [12]. After evaluation of the values of $Z(m_j)$ that storage module is selected for which $Z(m_j) = Z(m_j)_{min}$.

If there are no storage modules with the required parameters in the set, the values of the obtained parameters are taken as the initial requirements during developmeny of those models.

7. If especially rigid requirements are set for the storage system with respect to such parameters as cost, reliability, dimensions, weight, etc, the values of the main parameters of a hierarchic storage system obtained as a result of calculation by the given procedure, and also its architecture and structure, are taken to be only preliminary, which can be determined more precisely during the digital computer simulation of the hierarchic system as a whole.

Numerical example. Presented in the table are the parameters of multilevel storage systems, calculated according to the above procedure for various values of parameter α, that is, for practically different address tracks (for different algorithms). In that case the following values of the main initial parameters were Taken: $\beta = 0.6$, $C_n = 500 \cdot 10^3$ bytes, $S_0 = 5000$ rubles, $t_1 = 40$ nanoseconds. Since normalized units of storage capacity and cost were taken in the above expressions, then, according to [4] $F_0^{1/\alpha} = 50$, $b_0 = 5 \cdot 10^{-4}$. Then $C_n = 10^4$ and $S_0 = 10^7$.

Such storage systems can be realized only on storage modules of three types: of the static type on bipolar large-scale integrated microcircuits (BIS) with a capacity of 1K bytes and an access time of 0.04 microsecond, of the static type on bipolar BIS with a capacity of 4K bytes and an access time of 0.2 microsecond, and of the dynamic type on MDP-BIS with a capacity of 16K bytes and an access time of 1.8 microseconds.

Let us estimate the influence of errors in determination of the initial parameters included in the above expressions on the final result, that is, on the structure and main characteristics of the hierarchic storage system.

It should be noted above all that the parameters S_0 and C_0, appearing as limitations in the solution of optimization problems, are set by the purchaser in the technical task for development. The parameter t_1 can be determined without substantial errors from the condition of optimum matching in working time of the processor and the first level of the storage system. Therefore the greatest influence on the result is exerted by errors in determination of the parameters α and β, since they, firstly, are determined in the stage of storage system synthesis and, secondly, are present in the above formulas as exponents. Since it proved practically impossible to obtain in general form expressions for estimating the influence of errors in the

*By the truncated set $\theta^{(1)}$ we will understand the set of storage modules whose parameters are within the allowable limits determined by the values of the parameters obtained in point 6 of the procedure and the values of the parameters stipulated in the technical task.

determination of α and β on the values of parameters of a hierarchic system (in view of the complexity of the starting formulas), such estimates have been made on the basis of calculations of the parameters of hierarchic storage systems during variation of α and β in a wide range (α = 0.1 - 0.90 and β = 0.1 - 0.8 with discreteness of variation of 0.1 for each parameter). Analysis of the obtained results permits drawing the following conclusions.

Error in the determination of parameter α of over 50 percent leads to the addition of one level of the hierarchy, to change of parameters of the first-level storage capacity and the mean time of reference to the storage system by about 60-70 percent. Error of determinatin of α of less than 30 percent has no substantial significance, as for each level of the hierarchy a specific storage module is selected on the basis of values of parameters which satisfy the following inequalities:

$$C_{l(M)} > C_l, \quad t_{l(M)} < t_l, \quad b_{l(M)} < b_l.$$

Error in determination of parameter β has practically no influence on changes in the number of hierarchy levels, but change of β by 40-50 percent or more can lead to reduction of the first-level storage capacity and correspondingly to reduction of the productivity of the system by several tens of times. This is quite natural, since in essence parameter β reflects the level of development of the technology, that is, at what cost realization of a unit of storage capacity and a unit of speed is achieved.

Therefore, to exclude maximum errors in calculating storage system parameters, in the construction of the function b(t) and the determination of the value of parameter β it is necessary above all to use as a basis the parameters of specific storage modules from a set of modules available to developers.

Since the error in determination of parameter α and the limits of its variation affect the architecture and structure of the storage system, and also the effectiveness of application of that system, it must be recalled that the exact value of that parameter can be obtained only for a specific address track [10], that is, only for the specific algorithm. It is natural that the application of a planned storage system to solve other problems will be less effective. In that case the labor intensiveness of determination of parameter α by simulation, for example, according to the procedure described in [10], is smaller by more than an order of magnitude than that of simulation of the entire hierarchic storage system, for example, according to the procedure described in [13]. In the determination of parameter α in practice it is required only to determine the number of entries into the storage with the capacity C_1 during reference for information.

All the remaining parameters of the storage system are determined according to the presented formulas, which give results with acceptable precision, as the rounding off of values of the parameters obtained by calculation to the values of parameters of specific storage modules of the set of modules is at times so large that it becomes inadvisable to obtain more precise results.

In conclusion it should be emphasized that the presented procedure can be applied very effectively in the development of storage systems of specialized and problem-oriented computer hardware.

A Наименование параметров системы памяти	B Значения параметров системы памяти по уровням при соответствующих коэффициентах					
	$\beta=0,6$			$\beta=0,8$		
	$\alpha=0,2$	$\alpha=0,4$	$\alpha=0,6$	$\alpha=0,2$	$\alpha=0,4$	$\alpha=0,6$
1 1-й уровень:						
C_1К байт a	1,90	0,80	0,50	0,08	0,04	0,03
t_1 мкс b	0,04	0,04	0,04	0,04	0,04	0,04
b_1 тыс. руб c	0,53	0,22	0,15	0,70	0,36	0,30
2 2-й уровень:						
C_2К байт a	500,00	22,90	4,00	500,00	10,90	1,40
t_2 мкс b	12,00	0,98	0,20	214,00	9,79	1,53
b_2 тыс. руб c	4,46	0,92	0,42	4,30	1,12	0,60
3 3-й уровень						
C_3К байт a	—	500,00	42,00	—	500,06	30,46
t_3 мкс b	—	15,00	1,81	—	279,00	28.70
b_3 тыс. руб c	—	3,80	1,17	—	3,50	1,20
4 4-й уровень:						
C_4К байт a	—	—	500,00	—	—	500,00
t_4 мкс b	—	—	20,40	—	—	379,00
b_4 тыс. руб c	—	—	3,25	—	—	2,90
T мкс b	4,76	1,60	0,57	91,00	32,70	12,00

A -- Storage system parameters
B -- Values of storage system paremeters by levels at corresponding coefficients
1 -- first level
2 -- second level
3 -- third level
4 -- fourth level
a -- bytes
b -- microseconds
c -- 1000 rubles

BIBLIOGRAPHY

1. Artamonov, G. T., and Brekhov, O. M. Analiticheskiye modeli funktsionirovaniya EVM (Analytical Models of Computer Functioning). Moscow, Energiya, 1978, 368 pages.

2. Gecsei, I., and Lakes, I. A. A model for the evaluation of storage hierarchies. IBM SYSTEMS JOURNAL, 1974, Vol 13, No 2, pp 163-178.

3. Chow, C. K. On optimization of storage hierarchies. IBM JOURNAL OF RESEARCH AND DEVELOPMENT, 1974, Vol 18, No 3, pp 194-203.

4. Chow, C. K. Determination of caches capacity and its matching hierarchy. IEEE TRANSACTIONS ON COMPUTERS, 1976, Vol C-25, No 2, pp 157-164.

5. Zhuravlev, Yu. P., and Akin'shin, Yu. S. Sistemnoye proyektirovaniye pamyati TsVM (Systems Planning of Digital Computer Storage). Moscow, Sovetskoye radio, 1976, 344 pages.

6. Kartsev, M. A. Arkhitektura tsifrovykh vychislitel'nykh mashin (Digital Computer Architecture). Moscow, Nauka, 1978, 296 pages

7. Malinovskiy, B. N., Yakovlev, Yu. S., and Makovenko, Ye. T. Analiz i sintez mini-EVM (Mini-computer Analysis and Synthesis). Kiev, Naukova dumka, 1976, 174 pages.

8. Yurovskiy, B. Yu., Dem'yanovskiy, A. P., and Kharitonenko, A. A. Method of optimization of the storage hierarchy of computers according to a vector criterion. AVTOMATIKA I VYCHISLITEL'NAYA TEKHNIKA, 1978, No 1, pp 71-74.

9. Yakovlev, Yu. S., Makovenko, Ye. T., and Babyak, B. P. Distinctive features of the application of semiconductor storage devices in micro-computers. In the book: Tekhnicheskiye sredstva mini- i mikroEVM (Mini- and Micro-Computer Hardware). Kiev, Institute of Cybernetics, UkrSSR Academy of Sciences, 1979, pp 13-23.

10. Mattson, R. L., Gecsei, I., Slutz, D. R., and Traiger, I. L. Evaluation techniques for storage hierarchies. IBM SYSTEMS JOURNAL, 1970, Vol 9, No 2, pp 78-117.

11. Maninovskiy, B. N., Yakovlev, Yu. S., and Yurasov, A. A. Hierarchic storage system selection for mini-computers. USiM, 1975, 1975, No 3, pp 40-46.

12. Yakovlev, Yu. S., and Novikov, B. V. One approach to the selection of a modern elementary base for mini-computers. AVTOMATIKA, 1974, No 5, pp 68-74

13. Kuz'min, B. P. Hierarchic storage devices. In the book: Khraneniye informatsii v kiberneticheskikh ustroystvakh (Information Storage in Cybernetic Equipment). L. P. Krayzmer, editor. Moscow, Sovetskoye radio, 1969, pp 79-100.

[54-2174]

2174
CSO: 1863

SOFTWARE

UDC 681.3.06

OS-NTs-1 SYSTEM SUPERVISOR RUN ON ELEKTRONIKA NTs-1

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 59-64

[Article by Yu. A. Basin, engineer, Special Design Office of Computer Technology, Pskov Radio Parts Plant, Pskov]

[Text] Introduction. The supervisor is the nucleus of an operating system. Accomplishing centralized control, a supervisor combines the remaining parts of the system into a single whole [1]. As a nucleus a supervisor is realized on the zero level of the OS-NTs-1 system, representing means of organizing the computer process for all external levels [2,3]. The main functions of the supervisor are as follows:

--organizing the joint work of programs;
--providing means of communication and synchronization between processes and physical equipment;
--processing reports by which processes are interchanged;
--starting and conclusion of work of peripherals;
--processing interruptions from equipment [1].

The supervisor was developed in such a way that, on the one hand, it was a minimum part of a system capable of working independently and, on the other, it assured modular functional ability of a hierarchic structure to grow [3] and capability of parametrization for construction of an operating system with the required configuration. The basic principles of realization of a supervisor on the Elektronika-NTs-1 [4] computer and its modification the Elektronika-NTs-2 are examined in the given work.

Organization of the computational process. The requirements for organization of a computational process flow from the possibilities of the E-NTs-1 and E-NTs-2 computers and the proposed designation for use in complex real-time information-control system, characterized by developed control of the processes and data. From the point of view of control any real-time system can be represented by an aggregate of interacting processes, for the control of which the following questions must be solved:

--organization of the multiservicing of processes to assure independence of their performance and very full loading of the processor;
--determination of the form of representation of processes in operating systems in order to minimize the time and program facilities for their realization and control;
--connection with external events and synchronization.

The solution of these questions is based on selection of a unit of control and planning during division of the resources of the system. The task is selected as such a unit in the OS-NTS-1. The task is the program or aggregate of programs identified by a descriptor, briefly, the program state, and having independence in the sense of designation to a processor. In such a definition the problem most completely corresponds to the characteristics of the process and is the basic form of its realization. The task as the realization of a process can be taken for servicing of the system only as a result of external events generating the process. For that purpose, in the system initiation of the problem is assured from external events recorded in a register of computer interruptions and including in themselves the following groups: timers, sources on the immediate-access memory main line, processors and sources on the input-output unit main line.

In addition, provision is made for the initiation simultaneously of several problems from different groups of events. This permits increasing the reactivity of the system in real time. Initiation of the problem means that the descriptor of the corresponding problem is set in the queue of problems to be solved. The processing of interruptions has priority and is done on the basis of the general-system stack of external interruptions.

During the control of problems a multiproblem regime is realized in the system. The aggregate of problems ready for designation to a processor forms the queue of problems to be solved. Problems are serviced on a priority basis. The problem priority is given by a number from 0 to 31 and can be varied dynamically by the user.

The stack of external interruptions and the queue of problems to be solved form two lines of work for a processor. Recorded in the stack are the program state and the supervisor parts formed as a result of built-in external interruptions. The stack is initially analyzed upon designation of the next work for the processor. If it is not empty on the processor a program state addressed by the stack indicator is selected. When the stack is empty, an analysis is made of the problems to be solved, from which the problem with the highest priority is designated for the processor. The two queues are introduced from considerations of the effectiveness of servicing.

The problem is controlled in accordance with its descriptor and program state. The descriptor characterizes the problem as an independent unit of control and planning of resources and reflects changes of states of the problem during system servicing. The queue of problems to be solved, as other queues of problems, is constructed of problem descriptors. The descriptor structure is described below. The program state characterizes the problem as a unit of work on a processor and reflects the state of the processor with respect to the given problem.

The process accomplished by the problem can consist of a number of dependent "small" processes, the representation of which in the form of separate problems often is unwieldy or unacceptable as regards time. In that case the process can be realized in part of the problem. Such a process is controlled with respect to the program state on the basis of the personal stack of the problem, and the descriptor is common for all the component processes. Upon emergence of the program state process the corresponding parts of the problem are recorded in the stack and the designation for solution is made according to the rule of behavior of descriptors in the queue of problems for solution. During such control of "small" processes, expenditures on descriptor control are minimized, expenditures which can be commensurable with those for accomplishment of the process. Expenditures of storage on descriptors are reduced. Expenditures of storage on the program state are not substantial, as in most cases only the program state of initiated processes must be stored in the stack.

The realization of processes by the above-indicated methods presents the user with flexible control apparatus for the planning of practical systems. Possible here is the construction of systems with centralized control when a special task is distinguished which organizes the coupling and synchronization of processes. Decentralized construction of systems is thus assured.

The set of supervisor primitives performing the indicated functions of control of a computational process forms a task dispatcher, an inseparable part of any multitask system. Among the distinctive features of the task dispatcher under consideration is the short time required for switching work on the processor and flexibility of its control; a time for switching work on the processor of 20-80 microseconds; a mean time of installing the problem in the queue of problems for solution of 90 + 5n microseconds, where n is the volume of the queue.

In addition, the supervisor provides the possibility of description by the user and inclusion in the supervisor of its own software for processing external interruptions, extracodes and input-output device drivers not described in the system. Those facilities assure fairly effective adaptation of operating systems to systems of practical use.

Comment on queues [5]. A queue consists of elements in which descriptors of requests for corresponding operating system subsystems are contained. All queues in a system are organized on the basis of chain (coupled) lists. In each queue element there is a field with an indicator of the start of the queue. In the last queue element the communications field is a zero field. Physically the queues are arranged in tables divided into lines. Each line contains one queue element. The line format is constant for a given table. Used for work with queues are operations of installation in a queue, exclusion from a queue and selection from a queue. The installation in a queue is done either according to priority at any place in the queue during organization of servicing or at the start of the queue during organization of the stack (except problem queues). In all system tables the lines have an identical format, and the queue elements a complex structure, as a result of which the procedures accomplishing operations with queues are common for all tables and queues.

Each table has a supplementary list, formed of free lines of the table. Elements of that list carry no information, but serve as a reserve for the disposition of new descriptors during installation of requests in the main queues. After the descriptor has been rewritten on a line, no physical displacements of information are produced. The introduction of a list of free elements into the tables permits solving two important problems:

--the generation of tables with the required volume as a function of the equipment resources, the volume of the immediate-access storage and the needs of users, that is, parametrization of the system;

--dynamic consideration of the rate of use of table lines in the process of functioning of the system.

An element addressed by the request queue start indicator to the subsystem functions in the queue. Access to the working element is accomplished by selection from the queue, which consists in reading the queue start indicator. As a result of such organization of the control data, rapid access to the request descriptor is assured, as well as determinability of the control situations during the servicing of requests, that is, any effects and results of work of a subsystem relate only to an element addressed by the indicator of the start of that queue.

All the above also is valid for problem queues. The difference is that problem queues are organized on the basis of reserve (double combined) lists where there is a connection not only with a subsequent but also with a preceding element. The need for reserve lists occurs because the functioning of problems is closely connected with external events, signals from which can arrive for the commutation of problem states, generally speaking, at any moment of time. In that case the problem can be in an arbitrary state. Consequently, for searching operations the use of reserve lists gives essential minimization and convenience. The problem descriptor occupies eight storage cells and includes direct and reserve communication addresses, the problem priority, the problem program state stack indicator, and reference to technical data and service information.

In planning the processor time and subsystem resources distribution the highest priority is serviced first. For a queue on a processor the first request is displaced by one with a higher priority, and that increases the reactivity of the system in real time. For queues for subsystems, displacement of the first request is not allowed, as subsystems as resources allow only separate use. For elements with identical priority the rule "first come, first served" applies.

The structure of the control data and the queue servicing discipline determine the dynamic characteristics of the system as a whole. Therefore a considerable portion of the work in planning the system was done on analysis of the data structure and their standardization and unification. In the present version of the system the primitives of control data processing were realized on the program level.

The structure of the OS-NTs-1 system. In the examination of the structure of the OS-NTs-1 system [1] in this work attention should be turned first of all to the structure of separate levels of the system hierarchy. Solution of those questions is entrusted to the supervisor (the zero level) and determines to a great extent the effectiveness of the system.

The supervisor accomplishes centralized control in the system, and so transitions between levels (communication of subsystems) are accomplished through it. Presented is an approximate diagram of transitions in the system during initiation of the user program with representation of the corresponding data base. The functional interaction of the processes of the levels (shown on the diagram by a hatched line) is the sequence of processes in the system.

The processes of initiation (state 1) form the task for the system and transmit control to the given procedure on the level of data-segment control (state 2). The

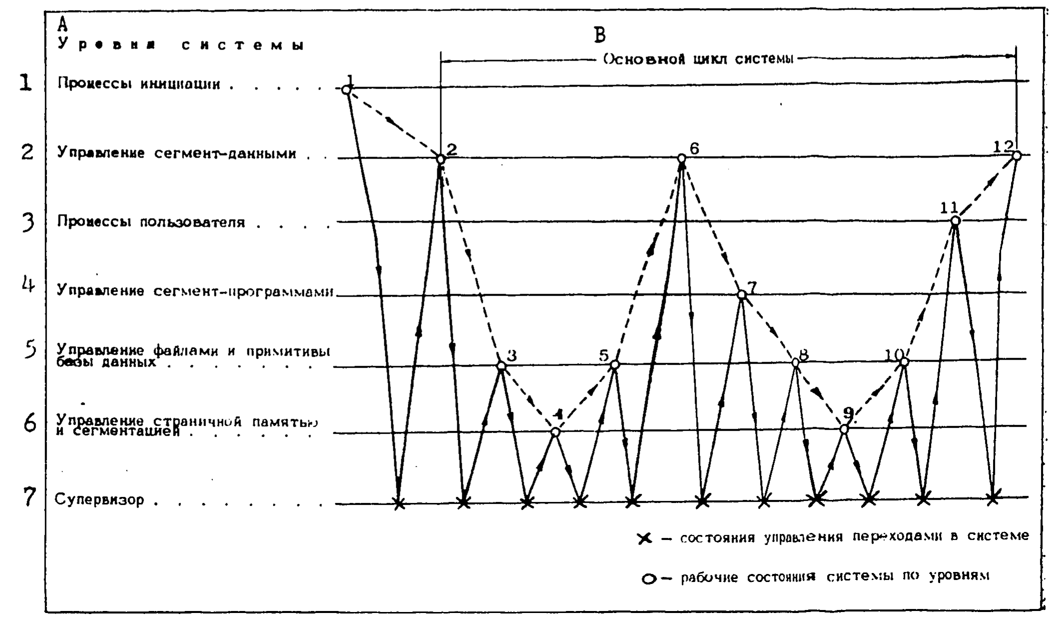


Diagram of transitions of states of a system by levels.

A -- System levels
B -- Main cycle of system
1 -- Processes of indication
2 -- Data-segment control
3 -- Processes of user
4 -- Program-segment control
5 -- File control and data base primitives
6 -- Control of page memory and segmentation
7 -- Supervisor
x -- States of transition control in system
o -- Working states of system by levels

procedure of data-segment control forms access to the data in accordance with the accomplishment of its algorithm (states 3-6), establishes a connection with the user process (states 7-10) and transmits control to it (state 11). The user process functions with an already known data base and when its processing is complete establishes connection with the level of data-segment control to present it with a new data base (state 12), after which the above-described series of processes is repeated, forming the main cycle of the system.

The diagram demonstrates the commutation of system states during transition between levels by means of a supervisor (shown by a solid line on the diagram). In contrast with hardware, such commutation conditions are not severe for the software and testify to joint use of a considerable part of the supervisor software by all the rest of the system levels. However, the same circumstance presents definite requirements for the organization of interfaces between levels. To solve that problem, a two-level organization of operating system subsystems is realized in the OS-NTs-1 system and operating system subsystems are formed as problems. The former assures the unification of subsystem interaction, and the latter a hierarchic structure of system construction.

In the system a class of work called external work is distinguished, one characterized by a law of control single in form. This can include, for example, work done in subsystems for the control of input-output, storage, files and the data base, etc. In all those subsystems the rule for servicing requests and synchronization of processes is identical. During corresponding unification of the control data, primitives of work control common to all the subsystems are obtained. The primitives for control can be separated from those for the performance of work and combined in the external work controller.

The external work controller represents a set of supervisor primitives which accomplish input into and output from the required subsystems. It also is intended for the formation of requests and their establishment in a queue to the subsystem in order to manage a queue of requests and synchronize processes formed by the subsystem, with a task, which presented a request for servicing of the subsystem.

The separation of external work in the supervisor assures a hierarchic two-level construction of the operating system subsystem. On the lower sublevel control over the system is organized, as well as communication with the system by means of the external work controller. Work of the subsystem is done on the upper sublevel. The lower sublevel control is accomplished on the level of the supervisor, with which a transition is accomplished to the actual level of the system during initiation of primitives of the subsystem upper sublevel. The algorithm for control and connection of subsystems with the system realized by the external work controller is the only one for all levels lower than the use level.

The principle of separation of functions of work control and performance permits considerably simplifying and formalizing the construction of separate subsystems and their connection with the system.

From the point of view of productivity of a system with centralized control in the supervisor under consideration the relative overhead expenditures are reduced because along with organizing connections between levels of the system the supervisor performs a portion of the work on controlling the work of the level.

A characteristic feature of the subsystems in which the work is classed as external is that they permit only separate use and their functioning is connected, as a rule, with change of work on the processor. Since a unit of work on the processor and a unit for which the resources are planned is a task, in the system it is assumed that the external work as resources is accessible only to tasks. This simplifies the connection and synchronization of processes and simplifies the structure of the subsystems. As a result of the above it follows that the subsystems using external work or subsystems, the functioning of which involves change of work on the processor, must be formulated as tasks. In such subsystems the primitives of the upper level are formed as tasks before the work is started. For that purpose a descriptor of the corresponding system problem is introduced into the problem controller. The formed system problem is establihsed in a queue and is serviced as an user task. All system service is accessible to system problems as well as user problems. The system problem is eliminated upon completion of execution.

The convenience of introduction of system problems lies in the fact that it is not necessary to acquire additional resources for synchronization of system processes. In addition, modular hierarchic growth capacity of the system with the use of corresponding levels of software as virtual machines is assured in a natural manner [3,6]. The formation of operating system subsystems in the form of problems determines their arrangement on the corresponding levels of the system. The apparatus of system problems imposes no sort of limitations on the system structure, besides multilevel structures with an unlimited recursion of transitions between levels.

A means of reference to subsystems is extracode apparatus, represented in the system of computer instruction with an array of parameters. Indicated in the array of extracode parameters is the subsystem to which reference is made, and the actual parameters are transmitted. Below, a subsystem for the control of physical input-output is presented as an example of the organization of an operating system subsystem.

Organization of input-output. Physical input-output is controlled on the supervisor level. In accordance with the organization of external work adopted in the system, the input-output subsystem has a two-level structure. Both the sublevels are arranged on the supervisor level. On the lower sublevel all actions are done by the external work controller on the control of requests for input-output units and for connection with the upper sublevel and the synchronization of input-output operations with tasks. Primitives of the upper sublevel--input-output unit drivers --perform operations directly connected with input-output organization. In that case the drivers not only control the input-output units in according with their specifics and automate the fulfilment of preparatory operations but also assure transormation of the interface of the input-output unit main line into an internal interface of the system, freeing the external levels from the details of input-output work and distinctive features of the main-line interface (the input-output main-line interface is given by a microprogram and can be different for different input-output units). Together with the driver the input-out unit represents a logical device for which only data exchange operations have been determined. The driver controls all input-output units of the same type in the system. The inclusion of a new input-output unit requires introducing an input-output unit of a corresponding driver into the subsystem. It should be noted that the driver does not use distributed resources but controls them, and working on the processor does not change during its functioning. As a result of that the input-output unit drivers are formulated as tasks, and communication with the lower sublevel is accomplished directly. This is still more correct because the two sublevels of the input-output subsystem function on one and the same zero level, where all is known to the supervisor.

Conclusion. The supervisor is constructed on the principle of division of functions among separate programs, which reduces the complexity of each of its parts [1]. Unification of the control data makes it possible to simplify the connections between programs and increase the reliability of the supervisor.

The principle of the division of functions of control and performance of work considerably orders the process of subsystem development, reducing the complexity and total volume of programs. This principle can be applied in the construction of multiprocessor systems with functional specialization of separate processors, and also during development of programmable controllers of peripherals.

Investigation of a system in accordance with the diagram permits simplifying the construction of a mathematical model and analysis of the dynamics of the system.

The developed supervisor occupies a volume of 3K 16-digit words. At the present time the OS-NTs-1 system supervisor has been debugged and is undergoing operation testing.

BIBLIOGRAPHY

1. Tsikritzis, D., and Bernstayn, F. Operatsionnyye sistemy (Operating Systems). Moscow, Mir, 1977, 336 pages.

2. Basin, Yu. A. Architecture of the OS-NTs-1 system. USiM, 1979, No 3, pp 55-57.

3. Dijkstra, E. W. The structure of the "THE" multiprogramming system. CACM, 1968, Vol 11, No 5, pp 341-346.

4. Shelikhov, A. A., and Selivanov, Yu. P. Vychislitel'naya tekhnika: Sbornik (Computer Technology: Collection). Moscow, Energiya, 1978, 216 pages.

5. Atkinson, T. D., Gagliardi, U. O., Raviola, G., and Shvenk, Kh. S. Architecture of modern central processors. TIIER, 1977, No 6, pp 37-46.

6. Gagliardi, U. O. Tendencies in the development of computer systems architecture. Ibid., pp 31-36.

[54-2174]

2174
CSO: 1863

UDC 62-52:681.3.06.2

'INTERFACE' PROGRAMMING LANGUAGE FOR ELEKTRONIKA-100

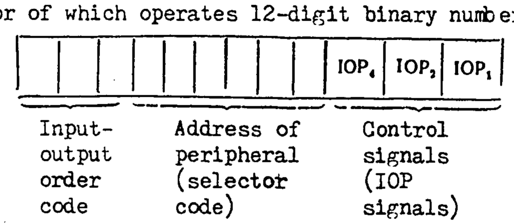
Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 89-93

[Article by G. V. Yukhimenko, senior scientific associate, Kiev Polytechnic Institute, Kiev]

[Excerpts] The problem-oriented programming language INTERFACE developed by the author, intended for description of the structures of the input-output interface circuits of peripherals connected to minicomputers with main-line organization of the interface, permits eliminating those shortcomings and assures ready adaptability of the simulating system in which it is used to various configurations of the peripheral simulating the computer.

In the present paper the INTERFACE language is described with reference to realization for a modeling system intended for the debugging of programs of a simulated minicomputer of the type of the Elektronika-100, Elektronika-100I, Saratov-2 and PDP-8. Minicomputers of the given types have a main-line organization of the interface and are equipped with a so-called combined channel of information exchange with peripherals, which consists of a program exchange channel in which the data exchange (one word per cycle of performance of an input-output order) is controlled by a special program, and the direct memory access channel, where data exchange is completely controlled by the interface circuit of a high-speed device, occupying cycles of a minicomputer central processor.

Before describing operators of the INTERFACE language, let us examine the structure of an input-output order and the main interface lines of the simulated computer, the central processor of which operates 12-digit binary numbers with a fixed point:



The structure of interface lines of a simulated computer is represented on the drawing (the number of communication lines is stated in brackets and the arrows show the direction of exchange).

As is evident from the figure, programmed exchange of data occurs over a single common bus, that is, all the peripherals are connected to interface lines of the processor simultaneously, but only the device whose address is indicated in the executed input-output instruction interacts with it. The direct storage access channel has separate address lines for determining storage cells during exchange and data lines, which assure direct access of the input-output unit to the processor storage without destroying the contents of its internal registers participating in accomplishing the main program. In that case, to assign the address of the storage cell and the counter of transmitted words it is possible both to use processor storage cells (three-cycle exchange, that is, during exchange the peripheral occupies three cycles of the central processor) and to realize them by means of registers of the input-output interface circuit (single-cycle data exchange).

A description is given of operators of the INTERFACE programming language in the Backus-Naur notation.

The INTERFACE language translator has been realized on a YeS-1020 universal digital computer in a disk operating system medium. As a result of the translator's work, programs of models of the input-output interface circuit of simulated peripherals are generated in the YeS EVM ASSEMBLER language. After subsequent translation those programs are readily included in the simulated system, realized also on the YeS-1020 and used for the debugging and optimization of large program complexes created on the basis of a mini-digital computer of the type of the ELEKTRONIKA-100.

The given simulating system and the INTERFACE programming language were used and showed high effectiveness in the development and debugging of software of a system for automated monitoring of the parameters of passive parts of hybrid LSI, which included a translator of the language of descriptions of the working monitoring programs and a control program intended for the control of four "GIPRID-P" measuring devices from a single ELEKTRONIKA-100I computer in a multiplex regime. It should be noted that the INTERFACE language can also be used in other simulating systems for the debugging of micro- and mini-computer programs with an analogous interface structure. Changes are required only for operators of description of an algorithm of action in a selector due to differences in the structure of the input-output instruction.

[54-2174]

2174
CSO: 1863

UDC 65.011.56:681.3.3.06

AUTOMATION OF PROGRAMMING FOR PATTERN GENERATORS AND THE PROBLEM OF COVERING PLANE AREAS WITH RECTANGLES

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 43-45

[Article by L. V. Nosov, Ye. G. Nosova, Ye. B. Rabinovich, engineers, and V. Z. Feynberg, candidate of physical and mathematical sciences, of "Gorizont" Production Association, Minsk]

[Text] At the present time pattern generators or, in another terminology, microphototypesetting devices, are widely used in automated systems of machine planning and manufacture of photographic integrated circuit patterns. In addition, pattern generators are one of the main elements of such systems and in essence the main task solved within the framework of automated systems for the manufacture of photographic integrated circuit patterns consists in automation of the obtaining of steering programs for pattern generators, that is, the task of constructing an automated programming system for pattern generation.

Let us examine the possibilities of pattern generators in general features on the example of the Soviet EM-519B, EM-549 and EM-539A systems. Those pattern generators are intended for the manufacture of precise intermediate originals (photographic patterns in the production of semiconductor and hybrid integrated circuits and are equipped with a positional system of numerical programmed control on a control minicomputer. The EM-519B and EM-549 pattern generators work on the principle of a single phototypesetter, according to which complex geometric patterns are exposed in sequence by rectangular typesetting elements, the dimensions, angle of inclination and position of which vary in accordance with the input information. The exposure of rectangular areas arranged at equal distances can be done in a cycle. The EM-539A pattern generator permits, along with the phototypesetter, accomplishing phototypesetting and photomontage. In group phototypesetting a topological pattern is exposed by matrices of rectangular elements (rasters), the number and dimensions of which can vary. The transition from one regime to another is accomplished automatically in accordance with the information fed to the pattern generator.

The problem of programming automation for pattern generators is now being solved within the framework of systems for machine planning of integrated circuits, or within the framework of autonomous programming automation systems. The process of automatic preparation on a computer programs for pattern generators itself starts when information about the topological pattern of the integrated circuit is prepared by the system of machine integrated circuit planning or manually by the topologist designer. Therefore the programming automation system for pattern generators is the lower link of each system of machine planning of integrated circuits.

The main and typical components of programming automation systems for pattern generators at the present time are:

-- the language of description of topological patterns of integrated circuits, oriented toward consideration of distinctive features of real integrated circuit topology (see [4], for example);

-- a syntactic translator from such a language in some intermediate intra-machine form;

-- a complex of software and hardware for monitoring initial information for design technological limitations (see [5,6], for example);

-- a complex of software which performs the basic processing of the initial data in calculating the steering program, but without connection with specific equipment--a set of postprocessors which reprocess the data into a form perceptible to specific pattern generators [7].

The programming automation system also includes program complexes for information editing, the input of program automation system arrays, of libraries of standard fragments, etc.

The main task in complexity, importance and labor-intensiveness is the task solved by the programs mentioned in point 4. That task arises naturally during automation of the preparation of steering programs for pattern generators working on the principle of single and group phototypesetting and consists in the construction of an effective algorithm for covering a plane area with rectangles.

The present paper describes a subsystem for covering areas of exposure with rectangles (POEP), for which that problem is the main one. That subsystem is a part of the programming automation system for pattern generators constructed on the basis of a standard language of description of topological drawings of photographic integrated circuit patterns and realized within the framework of YeS disk operating systems.

The initial information for the POEP subsystem is the resulting array of a syntactic translator from a language of topology description. That initial array for the POEP subsystem represents a description of the topology in an intra-machine form. It consists in the description of fragments and chains of operators, to whose action the fragments are subjected. The descriptions of fragments consist of a description of layers, which in turn consists of a description of contours.

The fact that it is not the result of operator action on fragments that is represented in the initial array, but the operators and operands (fragments) themselves permits covering areas of exposure with rectangles in a local fragment regime. In that regime all fragments of the topological pattern are regarded as independent patters. Then the obtained rectangles are subjected to operator action. This permits, during work in a local regime, covering the areas of exposure of each fragment only once, and not as many times as they are encountered in the description of integrated circuit topology, and this greatly reduces the working time of the subsystem. On the other hand, the local regime also makes it possible to obtain an array of covering with formally the same structure as the initial array, but greatly different from it in the fact that all the contours in the covering array are rectangles. This permits, in analyzing only operator chains (and not the entire initial topological pattern), solving with respect to programs the question of separating the part of the topology which it is advisable to expose, using additional working conditions of the display generator (besides single microphototypesetting) (for example, the exposure regime in the cycle for pattern generators EM-519B, EM-539A and EM-519 and the regime of phototypesetting for pattern generator EM-539A). In the POEP subsystem that possibility is realized by switching the program for reducing the operator chains to a special form.

The adopted covering array structure permits using the POEP subsystem to obtain control programs for other microphototypesetting installations by the development of only additional postprocessors.

The local regime imposes the following limitations on the coding of the topological pattern in the language of topology description:

--each fragment and each image of it in a positive or negative initial topological pattern, regarded by the subsystem as an independent pattern, must be a positive (or negative) pattern;

--if a fragment includes the boundary contour of a certain area of exposure, all the boundary contours of that area of exposure must be a part of the given fragment.

If those limitations are not observed during the coding of the integrated circuit topology, the POEP subsystem must accomplish the covering in a global regime. In that regime the operator actions of obtaining the absolute coordinates of the topological pattern must be decoded first, and then the entire topological pattern is covered with rectangles.

The POEP subsystem produces a covering with rectangles of both a potitive and negative topological pattern (areas of exposure of a negative pattern are external areas in relation to areas of exposure of a positive pattern). In that case the following problems are solved: expansion or contraction of areas of exposure to a given value; reduction of operator chains to a special form, that is, to chains consisting only of the operators MATRIX and MULTIPLICATION, which assures an effective working regime and optimizes the steering program for the above-mentioned pattern generators, separation from the initial description of the topological pattern (in particular, of the combined topological drawing) of the part of it that is to be exposed).

It is important to note that the subsystem does not require explicit description of areas of exposure in the initial information, but determines them automatically, in parallel with the process of covering the topological pattern. Consequently, in the subsystem there is excessive software (as in [2,6,8], for example) for the construction of areas exposed in explicit form.

Indicated in the task for performance of the POEP subsystem are:

--the type of topological pattern (negative or positive);
--the covering regime (local or global);
--whether in the local regime it is required to reduce the operator chain to a special form;
--whether it is required to expand or contract the areas of exposure;
--the combination of which identified elements of the initial array is a part of the initial topological pattern to be exposed.

The task of creating a universal programming automation system for pattern generators assumes the presence of software realizing the covering algorithm with the following distinctive features:

--they are acceptable for singly and multiply coupled areas even without clear separation of the areas of exposure;
--they are effective for obtaining the covering of both positive and negative topological patterns;
--for real integrated circuit topologies they have a working time comparable with the sorting time.

For areas all the boundary segments of which are parallel to the axes of coordinates, the task of constructing such algorithms was solved in [3,9,10]. In the POEP subsystem an algorithm has been realized for the solution of that task in the general case, when the boundaries of the areas of exposure can contained inclined segments. The algorithm is based on the idea of preliminary sorting the segments of boundaries of areas of exposure of topological drawings of integrated circuits and their subsequent processing by levels.

The solution of the problem of expansion (compression) of areas of exposure consists in the expansion (compression) of the external and compression (or expansion) of internal contours bounding the areas of exposure. It is understandable that this problem can be solved as a linear number of operations with respect to the number of boundary segments, if the areas of exposure are given in explicit form. In the contrary case the algorithms presented in [2,6,8] can be used for their program determination, but that leads to the construction of additional arrays and complicates the subsystem. In POEP, to solve that problem in the case where areas of exposure are not clearly given, an alforithm is realized, one based on the same idea as the covering algorithm, and the same array of sorted setments, available as an initial array.

To assure effective functioning of the POEP subsystem in both global and local covering conditions the segment array is formed in the form of a linear list in the main computer storage and when the storage is overfilled continues to form further on magnetic disks. To sort that array in the storage, sorting by confluence on magnetic disks is used, and to sort it on magnetic disks a YeS disk operating system standard sorting program is used. Programs of covering and expansion (contraction) also are oriented toward work in both the main storage and on magnetic disks.

To cover lines of constant width (tracks), which are often encountered in real integrated circuit topologies, an algorithm has been realized which optimally covers the track as the linear number of operations with respect to the number of boundary peaks of the track. The use of that algorithm is stipulated by the fact that the track covering obtained by the basic algorithm, the boundary of which contains inclined segments, is far from optimal. The POEP subsystem is realized in the ASSEMBLER language in the YeS disk operating system. On the YeS-1020 computer the areas of exposure of a topological drawing of an integrated circuit consisting of 5000 circuits (a total of 50,000 peaks) was covered. The computer working time was about 15 minutes in a local covering regime, and about 80 minutes in a global regime.

BIBLIOGRAPHY

1. Glazkov, I. M. Oborudovaniye dlya proizvodstva fotoshablonov i vypolneniya operatsiy fotolitografii (Equipment for Production of Photographic Patterns and Performance of Photolithographic Operations). Moscow, 1975, 74 pages.

2. Stempkovskiy, A. L. Main problems of pattern generator software. ELEKTRONNAYA PROMYSHLENNOST', 1976, No 9, pp 365-372.

3. Sverdlov, A. G., and Feynberg, V. Z. Automated system for preparation of pattern generator programs. ELEKTRONNAYA PROMYSHLENNOST', 1978, No 10(70), pp 18-20.

4. Kazenkov, G. G., Batalov, B. V., Feynberg, V. Z., et al. Topology description language. MIKROELEKTRONIKA, 1976, No 9, pp 234-248.

5. Shepelev, V. A., Stempkovskiy, A. L., and Stepchenkov, V. ?. Complex monitoring of topology and photographic patterns. Ibid., pp 296-306.

6. Nosov, L. V., Rabinovich, Ye. B., and Feynberg, V. Z. Programmed monitoring of drawings of integrated circuit photographic patterns. ELEKTRONNAYA TEKHNIKA. SERIYA 3, 1978, No 4 (76), pp 95-102.

7. Feynberg, V. Z., and Edel'shteyn, A. I. Postprocessors of a system of programming automation for pattern generators. Ibid., SERIYA 7, 1977, No 6 (85), pp 78-84.

8. Feynberg, V. Z., Sverdlov, A. G., and Rabinovich, Ye. B. Automatic determination of areas of exposure of integrated circuit patterns. Ibid., SERIYA 3, 1978, No 1, pp 48-54.

9. Shiro, G. E., Osipov, L. B., and Volkov, V. I. Parallel rapid algorithm for data preparation for microphototypesetting installations. Ibid., SERIYA 11, 1976, No 3, pp 71-76.

10. Nosova, Ye. G., Sverdlov, A. G., and Feynberg, V. Z. Algorithms for division of plane patterns in systems for computer designing of integrated circuits. IZVESTIYA AN BSSR. SERIYA FIZ.-MAT. NAUK, 1978, No 5, pp 16-24.

[54-2174]

2174
CSO: 1863

UDC 681.142.62

LANGUAGE FACILITIES OF A DISPATCHER CONTROLLING PROBLEMS OF REAL-TIME OPERATING SYSTEMS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 51-56

[Article by V. I. Golovach and V. M. Sobol', engineers, Moscow]

[Excerpts] Introduction. The UZOR [Upravleniye Zadachami Operatsionnykh sistem Real'nogo vremeni--Control of Real-Time Operating System Problems] dispatcher serves for the development of real-time programs for mini- or microcomputers in the ASSEMBLER language. Macroinstructions which the user (the real-time program developer) can use equally with "real" ASSEMBLER microinstructions in a computer are called the language facilities of the dispatcher. They expand the instructions list of the computer and ease the programming of real-time problems.

System subroutines forming a part of the dispatcher correspond to dispatcher microinstructions; thus the language facilities completely determine the dispatcher functions.

Use of the UZOR dispatcher permits developing hierarchically ordered (structured) real-time programs, where the dispatcher subroutines represent the lowest level (nucleus) of the corresponding structure. The dispatcher includes only the drivers of real-time clocks (such clocks are an obligatory component of most modern computers). The necessary drivers must be programmed by the user himself by means of dispatcher language facilities. One of the most important features of the UZOR dispatcher is independence from peripherals.

The transition to the development of dispatcher language facilities was derived from the MODULA language [1], but the proposed language facilities differ from those of MODULA multiprogramming in the following dispatcher properties:

--the possibilities of dynamic creation and annihilation of processes;
--the application of other means of process synchronization;
--the need for designation of priorities for all processes;
--the presence of a mechanism of emergency situations, serving, in particular, for the debugging of real-time programs.

Most dispatcher language facilities are machine-independent and can be realized by means of macrofacilities included in the basic software of the corresponding computer.

The UZOR dispatcher has been realized for the SM-3 minicomputer and the "Elektronica-60" microcomputer.

The authors wish to express their gratitude to Yu. F. Gal', M. V, Koroleva and Ye. G. Stalin for useful discussions which to a great extent determined the essence of the present work.

BIBLIOGRAPHY

1. Wirth, N. MODULA: a language for modular multiprogramming. SOFTWARE--PRACTICE AND EXPERIENCE, 1977, Vol 7, No 1, pp 3-37.

[54-2174]

2174
CSO: 1863

UDC 681.3.06

DEVELOPMENT OF MULTIFUNCTIONAL OPERATING SYSTEMS FOR AGGREGATE SOFTWARE SYSTEMS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 57-59

[Article by A. B. Ayzenberg, senior scientific associate, Scientific Research Institute of Control Computers, Severodonetsk]

[Excerpt] ASVT-M M-7000 and SM-2 computer complexes until recently were equipped with a package of program modules for the composition of operating systems for aggregate software systems. The package permits creating operating systems very varied in their functional possibilities and hardware used, starting with very simple diskless single-purpose to real-time multiprocessor disk operating systems. Later, on the basis of a package of programmed modules for aggregate software systems a new alternative of the package was developed, by means of which it became possible to combine in a single operating system not only a real-time regime with package processing but also a real-time regime with package processing and/or a multipanel time-sharing regime. At present the SM-2 is equipped with precisely that package.

[54-2174]

2174
CSO: 1863

DESIGN OF DATA PROCESSING PROGRAMMING SYSTEMS

Moscow KONSTRUIROVANIYE SISTEM PROGRAMMIROVANIYA OBRABOTKI DANNYKH in Russian 1979 signed to press 20 Nov 79 pp 6-7, 79-81, 159

[Excerpts from book by Svetalana Nikolayevna Berestovaya, Ol'ga Leonidovna Perevozchikova, Vladimir Mikhaylovich Romanov and Yekaterina Logvinovna Yushchenko, Izdatel'stvo "Statistika", 15,000 copies, 272 pages]

[Excerpts] This monograph elucidates the problem of automating the process of designing programming systems as a whole. Furthermore, attention to questions of designing and of the technology for designing compilers from data processing languages is accentuated, including processors based on problem-oriented languages. This book is based on the experience of creating a number of familiar domestic and foreign systems, and primarily systems in the implementation of which the authors took direct part, such as the Cobol-Dnepr-21 software system, the Cobol-ASVT-M3000 software system, the DISPROM dialogue parametric programming system and the YaOD-75 dialogue system generated by it for the "Minsk-32" computer and designed for the solution of problems by nonprofessional users, and the RYeF problem-oriented system for solving on the M4030 computer problems in the design of ASU [automated control system] data software.

The book consists of two parts in six chapters. The first half consists of the first two chapters, in which a description is given of the key software of modern data processing languages and of such established languages as Cobol, PL/1 and RPG [report program generator], and descriptions are also given of certain problem-oriented languages. It should be emphasized that in this monograph as much as possible is used the terminology established in the State standard for the Cobol language, whose properties are used in many examples presented for the purpose of illustrating data processing processes.*

The second half of this monograph (chs 3 to 6) is devoted to methods and procedures for designing data processing systems. Chapter 3 is devoted to a discussion of classes of data processing systems. Furthermore, against the background of the problems in using them is disclosed the topicality of the utilization of data base technology in the implementation of data processing systems.

---

*"Yazyk programmirovaniya Kobol. GOST 22588-77." [The Cobol Programming Language; All-Union State Standard 22588-77], Decree of the USSR Gosstandart [State Committee for Standards] No 1372, 27 May 1977.

Chapters 4 and 6 in the second half are central. In these the authors attempted to fill in the gap existing in domestic literature in the area of designing and implementing data processing systems. In chapter 4 is given a detailed description of the major methods and algorithms for translating data processing languages; in chapter 6 are discussed the fundamentals of the parametric approach to designing language processors.

Chapter 5 serves the purpose of illustrating the application of the methods discussed in chapter 4. In it is described the implementation of four data processing systems.

It is the intent of the authors that this monograph serve as a practical textbook for a wide range of systems programmers involved in creating and using data processing systems. It is suggested that the reader have definite knowledge of the fundamental concepts of programming and programming languages, in particular, of the Algol-60, Fortran, Cobol and PL/1 languages which have become established and are used in the educational process at VUZ's.

Table 3.1.

Name of system and bibliographic reference	Basic system software	Development status	Type of system	Permissible data structures	Basic languages	Description of YaMD [data manipulation language]
IPS-32 [42]	"Minsk-32" supervisory program	In use	Closed	Serial files whose entries have a hierarchical tree structure	-	Set of operators of two types controlling the following: retrieval of data through complex circuits in the form of conditions and ranges of retrieval keys, and the formation of output files; renewal of data in files
INFOR [43]	"Minsk-32" supervisory program	In use	Closed	Serial files whose entries have a hierarchical tree structure	-	Production of reference information which is separated successively from various files by transforming them; renewal of files

MARK-IV [44]	YeS [unified series] computer DOS [disk operating system] and OS [operating system]	Introduced in Sep 1973	Closed	Serial and serial indexed files whose entries have a hierarchical tree structure	-	Tabular language by means of which are possible retrieval of data according to conditions, logical and arithmetic conversion of data and tabulation of the latter and ordering of files by keys
SIOD [23]	ASVT [modular system of computer technology] DOS	Industrial use of first phase of the system; the second phase will expand the ability to work with the system in real time	Hybrid	Entries having a hierarchical tree structure; relations between groups of entries are permitted	Cobol, ANSI-Cobol, Fortran and assembler	YaMD includes operators which enter data by key; accessing of data by key and condition, and also output of data with format instruction
VYeGA [45]	BESM-6 DISPAK OS	In use	Closed	Trees with hierarchical repeating groups	-	LIRS language designed for writing assignments consisting of descriptions of data with an indication of criteria for accessing them in documents and a set of operators which describe operations on documents
BANK [27]	YeS DOS, YeS OS; being expanded for ASVT DOS	In use	With basic language	Net structures	Assembler, being expanded for Cobol and PL/1	Language in the form of a set of macroinstructions for retrieving data by key and condition and for modification of data in the base

SINBAD [46]	YeS OS	First phase of system introduced; SINBAD-2 will include facilities for documentation and restoration	With basic language	Hierarchical tree structure; additional relations at group level also permitted	Assembler, Cobol, PL/1	Manipulation language in the form of a set of macroinstructions and designed for accessing data with assignment of the process of retrieval and renewal of the data base
OKA [24]	YeS OS	Industrial utilization of first phase of the system with a package processing mode; teleprocessing facilities are included in the system's second phase	With basic language	Ditto	Assembler, Cobol, PL/1	BETA language consists of a set of the following macroinstructions: for data (segment) retrieval; replacement of data; exclusion of data; and addition of new data to the base
NABOB [27]	YeS DOS	First phase in development	With basic language	Entries of variable structure consiting of repeating groups; relations established between entries	PL/1	YaMD is a set of macroinstructions for the retrieval and output of data, as well as for the modification of the data base

BAŹIS [25]	ASVT DOS	In use	Hybrid	Hierarchical data structures	Special basic language for describing data, Cobol, assembler	Dialogue language with orientation toward nonprofessionals makes it possible to retrieve and access data with subsequent output in a definite format
SIZIF [16]	ASVT DOS	Design stage	Hybrid	Hierarchical data structures		The problem-oriented YaMDV language is constructed with the use of a predicate algebra and makes it possible to manipulate relations on the basis of the operations of relational algebra; permissible types of relations are unary and binary upon hierarchically subordinate data
POISK [13]	BESM-6 "Dubna" monitor system	Being introduced	Hybrid	Net structures at the data representation level; model of data for the user--data relations	Autocode, Madlen (a language of the assembler type) and Fortran	The language offers nonprogrammer users the opportunity to operate with n-ary relations by means of operators similar to the following operations of relational algebra: product, combination, division, selection by condition, and projection

Chapter 5. Realization of Concrete Data Processing Programming Systems

This chapter is devoted to a brief discussion of several instances of the realization of automated data processing systems (ASOD's), which were selected for the purpose of an interpretation in real systems of the methods and procedures for designing programming systems described in chapter 4. In describing these systems the authors have strived to reflect the line of development of the principles and methods of designing ASOD's and of ideas regarding methods of solving problems with computers.

The first of the systems discussed, the Cobol-ASVT software system (1973), is a multi-pass compiler in whose design was employed the principle of modular programming. In view of this, in the description a rather detailed discussion is presented of the coprogram method and of the hierarchical principle of controlling modules. Let us note that chapter 6 is devoted to methods of designing long programs which have been given the names of modular and structural programming.

The second system discussed in this chapter, the Cobol-Dnepr-21 software system (1969), includes a classical syntactically controlled compiler oriented toward the class of input languages. The third system described in this chapter, the YaOD-75 (1975), is a dialogue programming system whose compiler is designed according to the step-by-step principle. The last system, the RYeF (1977), is designed for communication between users and an SUBD [system for control of data bases] and includes an optimizing translator of inquiries to the SUBD (references) to an intermediate high-level language.

[76-8831]

8831
CSO: 1863

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

MACHINE GRAPHICS FOR THE BESM-ALGOL SYSTEM

Moscow MASHINNAYA GRAFIKA DLYA SISTEMY BESM-ALGOL in Russian 1978 signed to press 14 Aug 78 pp 2, 8, 11, 174-175

[Annotation, author's foreword and excerpts from book by Vadim Valerianovich Kobelev, Izdatel'stvo "Nauka", 25,000 copies, 176 pages]

[Text] This book is devoted to a detailed description of the GRAFAL system--a system of machine graphics procedures in the Algol-60 language for BESM-6 computers. A discussion is presented of methods of forming and transforming graphic information to be processed on a graph plotter of the step-by-step type.

This book is intended for experimental programmers and can serve as instructions for practical work with the GRAFAL system. This book can also be of interest to a wide range of specialists in computer technology and to students and graduate students at higher educational institutions specializing in the field of systems programming and computer software.

Author's Foreword

The system of procedures described below makes it possible within the framework of Algol programs to work with CALCOMP-763 and -765 graph plotters connected to a BESM-6 computer.

Almost all procedures contain a text written in Algol autocode (cf. [1]). The use of the autocode makes it possible easily and conveniently to arrange a link between procedures and to increase their speed considerably. In addition, this considerably limits the range of application of the system of procedures described, since they prove to be tied to a specific type of computer.

The presence of an autocode in the fields of procedures imposes limitations on the use of index registers in the user's program, which can also contain an autocode. Index registers from 6 to 12_8 can be damaged in the operation of the procedures described. Therefore, there must not be procedures above the fourth steady-state level in the user's program.

The overwhelming majority of procedures described here apparently do not contain errors. They have been tested in numerous examples and normally operate with translator versions from 1 Oct 71, 5 Feb 73 and 9 Apr 74 and with D-78 and DISPAK supervisory programs.

The system of procedures described is rather extensive. Their number has reached 267. By analogy with the system of graphic procedures in Fortran [2], this system has been given the name GRAFAL (GRAFicheskiy ALgol [Graphic Algol]).

The originals of all figures contained in this book were made on a graph plotter and were transferred into typography without additional technical treatment.

The author considers it his pleasant duty to express his profound gratitude to V.S. Sukhinov, A.N. Tomilin, V.F. Tyurin, V.M. Yermakov, A.S. Levina, O.A. Chernenko and G.P. Moiseyeva for consultation, assistance and steady interest in this work.

2. Brief Information on the SP77

The SP77 program (standard program No 77) developed by V.S. Sukhinov and A.N. Tomilin transmits serially to the graph plotter for execution preformed instructions grouped into an instruction array. The SP77 processes about 400 instructions per second.

Bibliography

34. Bokova, I.D. et al. "Operatsionnaya sistema DISPAK dlya BESM-6 (pol'zovatelyu)" [DISPAK Operating System for the BESM-6 (User)], No I. Bokova, I.D., Zel'dinova, S.A., Zuyev, V.I., Koryakin, V.K., Koshkina, L.V., Ozornin, Yu.V., Tyurin, V.F. and Shulepov, N.I., Moscow, USSR Academy of Sciences Institute of Problems of Mechanics, 1973.

[76-8831]

8831
CSO: 1863

UDC 681.142.4

ALGORITHMIC MODELS IN THE AUTOMATION OF RESEARCH

Moscow ALGORITMICHESKIYE MODELI V AVTOMATIZATSII ISSLEDOVANIY in Russian 1980 signed to press 27 June 1980 pp 2-4, 247-249

[Annotation and table of contents from the book edited by Doctor of Engineering Sciences Professor V.M. Ponomarev, Izdatel'stvo "Nauka", 3,950 copies, 255 pages]

[Text] The collection is devoted to problems of automating research studies, and in particular, the design and utilization of algorithmic models of complex systems and processes. Algorithms and programs are described for the processing of large data files. The possibilities for the application of the modeling principles worked out here to the field of computer engineering, robot engineering, economic, ecology, information sciences and in the design of data base for items of cultural value are analyzed using specific examples.

The book is intended for a wide circle of scientific and engineering workers.

Foreword

One of the most important features of the contemporary scientific and engineering revolution is the rapid growth of the volumes of information being obtained and processed in practically all fields of science and engineering. The development of modern methods and tools for scientific and engineering research studies oriented towards the widescale utilization of computer equipment, brings about the necessity of designing fundamentally new forms of interfacing the research with the computer, especially, to the necessity of developing principles for the design and utilization of algorithmic models and automated systems. It is obvious that the choice of the model determines the depth of understanding of the phenomenon being studied to a significant extent. Because of this, various aspects of the utilization of algorithmic models and their potential capabilities for increasing efficiency and quality in planning, as well as the performance of scientific research are of considerable interest.

Under these conditions, it is quite important to treat the possibilities of the widescale application of general principles for the automation of reasearch in the most diverse scientific fields, including in such urgent fields having considerable significance for applications as planning, robot engineering, economics, the construction of data bases for items of cultural value, information sciences, ecology, etc.

The results of work in three main directions, being carried out at the Leningrad Scientific Research Computer Center of the USSR Academy of Sciences are presented in this collection.

Questions related to algorithmic problems of the automation of scientific research are treated in the first section, "algorithmic models"; the algorithmic models used in the process of solving complex multiple aspect problems in various fields of contemporary science and engineering are analyzed.

The results of work to design program libraries are presented in the second section, "data processing algorithms and programs", and algorithms are described which are intended for the performance of complex mathematical calculations and the processing of large masses of experimental data.

The principles for the design of robot engineering systems and their control, as well as ways of designing automated control systems for various purposes are treated in third section, "automated systems".

The urgency and newness of the problems treated here, and the possibility for the widescale practical application of the results obtained to various fields of science and engineering allow us to hope that the materials being published will be of interest to a wide circle of specialists.

Table of Contents

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

III. Automated Systems

[64-8225]

8225
CSO: 1863

UDC 681.3.01

SOFTWARE LIBRARIES AT THE LENINGRAD COMPUTER CENTER OF THE USSR ACADEMY OF SCIENCES

Moscow ALGORITMICHESKIYE MODELI V AVTOMATIZATSII ISSLEDOVANIYE in Russian 1980 signed to press 27 Jun 80 pp 66-68

[Article by G.I. Podol'skaya: "The State of the Art of Program Libraries at the Leningrad Scientific Research Computer Center of the USSR Academy of Sciences"]

[Text] At the present time, program software solves the problem of providing maximum conveniences to the user in interacting with a computer and when solving applied problems. One of the aspects which assists in the resolution of this problem is a library of scientific collective use routines.

The domestically produced BESM-6 computer and the American CYBER computer system are universal computer complexes. This also leaves its imprint on the library of standard routines which should satisfy a wide circle of users. The program libraries for the BESM-6 include the following major sections: linear algebra, the extraction of routes, the minimization of functions, ordinary differential equations, the computation of integrals, interpolation and approximation of functions, the representation of functions in the form of theories, equations of mathematical physics, special functions, the generation of pseudorandom numbers, the statistical analysis of information, optimal planning, the sorting and ordering of data files, operations with complex numbers, printouts on an alphanumeric printer and operation with peripherals.

The libraries of programs for the BESM-6, which are being used at the Leningrad Scientific Research Computer Center of the USSR Academy of Sciences, can be classified according to the names of the existing program systems, which operate within the framework of the DISPAK operating system or the DISPAK disk operating system:

1. The library of programs in FORTRAN and MADLEN;
2. The library of ALGOL routines;
3. The library of SSP programs (FORTRAN);
4. The library of the GRAFOR program system;

5. The library of Wilkinson routines;
6. The library of routines for numerical analysis (FORTRAN);
7. The library of routines in BESM-ALGOL.

The first library numbers about 350 routines and consists of two sections. The bulk of the library contains the most used mathematical and service routines, auxiliary (supplemental) routines for problems in nuclear physics, high energy physics and other specific programs. The majority of this program is written in FORTRAN, and about 40 algorithms are written in MADLEN. A specially designed programming mechanism in the "Dubna" monitor system is used for operation with the indicated library [1]. A specific feature of this mechanism is the fast that when accessing the general library, only programs needed for the solution of the given problem will be read into the temporary library.

The second library, which is in service in the Leningrad Scientific Research Computer Center of the USSR Academy of Sciences (LNIVTs), was created in the State Optical Insitute imeni Vavilov in conjunction with the Insitute of Social and Economic Problems of the USSR Academy of Sciences (ISEP). This library also contains routines for the major areas of mathematics as well as service procedures. It includes about 260 algorithms, which are regularly corrected and updated.

The SSP (Scientific Subroutine Package) problem library, designed by the IBM company, is basically oriented towards the solution of statistical problems, as well as problems of matrix and linear algebra. The library is written in FORTRAN and contains about 400 routines [2]. Because many algorithms were not checked when this library was translated for the BESM-6, the associates of the LNIVTs of the USSR Academy fo Sciences did considerable correction work, as well as substitution and testing of the programs. Since the operational time with this library is small in the LNIVTs of the USSR Academy of Sciences, information is being gathered at the present time on the use of algorithms, errors are being ascertained and recommendations and advice of the users are being taken into account.

The GRAFOR library makes a large set of standard routines available to the user for working with a graph plotter. The authors of the library are associates of the Institute of Applied Mathematics of the USSR Academy of Sciences, who are constantly supplementing and expanding this library.

The remaining problem oriented libraries mentioned above will be placed in service in 1979. These are: the library for numerical analysis methods (Computer Center of Moscow State University), containing about 300 routines and the linear algebra library (Wilkinson) [3].

One of the most complete and widely used is the library of routines in BESM--ALGOL. It contains about 360 algorithms and the authors are the associates of the ISEP of the USSR Academy of Sciences and Leningrad State University. In terms of its composition, this library basically matches the library of routines written in ALGOL. The library is being constantly corrected and supplemented.

The lack of a program mechanism which makes it possible to work with the requisite algorithm from any programming system, forces the duplication of the widely used program for various programming systems. This is also due to the large number of similar libraries for the BESM-6 computer.

The library of scientific routines for the CYBER computer system is devoid of this drawback, and the programs of this systems can be accessed from practically any programming system. The library contains about 400 routines in FORTRAN and assembler for the following areas: arithmetic; elementary functions; polynomials and special functions; ordinary differential equations; interpolation, approximation and squares; linear algebra; probability, statistics and time series; non-linear equations.

The libraries of routines enumerated above are widely used in the LNIVTs of the USSR Academy of Sciences. They substantially facilitate the work of users, make it possible to save time and offer considerable possibilities for the choice of the requisite algorithms in various areas of mathematics.

Work will be done in the future at the LNIVTs of the USSR Academy of Sciences to expand the already existing library and to create narrowly specialized packages of applied programs.

BIBLIOGRAPHY

1. Maznyy G.L., "Programmirovaniye na BESM-6 v sisteme'Dubna'" ["Programming with the BESM-6 in the 'Dubna' System"], Moscow, Nauka Publishers, 1978.

2. "Sbornik nauchnykh programm na Fortrane" ["Collection of Scientific Programs in FORTRAN"], Moscow, Statistika Publishers, 1974, Nos. 1, 2.

3. Wilkinson, Raynsh, "Spravochnik algoritmov na yazyke ALGOL: Lineynaya Algebra" ["A Handbook of ALGOL Programs: Linear Algebra"], Moscow, Mashinostroyeniye Publishers, 1976.

[64-8225]

8225
CSO: 1863

APPLICATIONS

UDC 551.1:550.34

PROBLEMS IN VIBROSEISMIC RESEARCH METHODS

Novosibirsk PROBLEMY VIBROSEYSMICHESKIKH METODOV ISSLEDOVANIYA in Russian 1979 signed to press 24 Dec 79 pp 2-6

[Annotation, table of contents and foreword from the collection of works edited by Anatoliy Semenovich Alekseyev and Aleksey Timofeyevich Gorbachev, Computer Center, Siberian Department, USSR Academy of Sciences, 500 copies, 173 pages]

/Text/ ANNOTATION

This collection of works is devoted to the problem of utilizing nonexplosive, mechanical sources of seismic waves to investigate the Earth's deep structure. The authors discuss questions relating to the development of multichannel seismic data collection and processing systems for the solution of problems in seismic surveying and deep seismic sounding.

This collection is aimed at the extensive circle of specialists working in the field of geophysics.

TABLE OF CONTENTS

FOREWORD

This collection of works contains several articles that were presented at the conference on methods for the vibration radioscopy of the Earth that was held in June 1979 in Novosibirsk.

The development of controllable mechanical sources of seismic waves that are capable of creating oscillations with repeatable forms is now of particular urgency because of the trend toward the use of areal multielement observation systems and a wave field's dynamic characteristics in the interpretation of observations that is now being seen in seismic surveying, methods for the deep seismic sounding of the Earth, and seismology.

This change is dictated by the development in the theory of seismic waves of methods for solving inverse dynamic problems, as well as the appearance of a number of important practical problems in seismic surveying, the solution of which requires high-quality dynamic measurements and two-dimensional observation systems.

In the area of petroleum seismic prospecting, among these problems we can include searching for nonstructural pools, studying the Paleozoic severely dislocated complex, searching for structures under trap formations, exploring subsalt structures, and a number of other geological surveying problems in which the three-dimensional nature of the investigated mediums and the necessity of using dynamics waves manifest themselves quite substantially.

In deep seismic investigations of the Earth's crust and in seismology such problems have arisen, in particular, in studies of the horizontal irregularities in the Earth's crust and mantle, as well as during the quantitative description of zones with reduced wave propagation speed values.

In such instances, the advantages of the use of controllable mechanical sources over the use of explosions and earthquakes for the organization of practical investigations are not only technological. One important advantage of these types of sources is the possibility of the sequential synthesis of multielement areal systems related to a stable source form, as well as the enlargement of the dynamic range of the measurements because of the correlational or cophasal storage of useful signals while simultaneously suppressing noncorrelated interference during extended emission sessions.

The increase in the volumes and quality of field information requires the use of new techniques for receiving and processing it. This field is covered by the articles in this collection in which the authors discuss questions on the creation of multichannel seismic stations, seismic receiving antennas and field processing complexes based on mini- and microcomputers.

Questions relating to the theory of areal systems and methods of interpreting dynamic measurements were discussed at the conference but are not included in this collection, since they were the subject of a special conference on "Numerical Methods for Processing Seismic Observations," after which an appropriate collection of works was prepared for publication.

On the whole, the materials in this collection can be of interest to geophysical specialists and engineers who are engaged in developing the principles of pulsed and vibration emission of seismic waves, as well as electronics engineers concerned with the problems involved in creating geophysical information collection and processing systems.

/67-11746/

11746
CSO: 1863

UDC 551.1:550.34

A FIELD COMPUTER COMPLEX FOR THE COLLECTION AND PROCESSING OF GEOPHYSICAL INFORMATION UTILIZING THE 'ELEKTRONIKA NTS-OZT' MICROCOMPUTER

Novosibirsk PROBLEMY VIBROSEYSMICHESKIKH METODOV ISSLEDOVANIYA in Russian 1979 pp 39-43

/Article by A.S. Alekseyev, A.T. Gorbachev, B.M. Glinskiy, V.M. Ivanov, V.M. Lobastov, M.I. Mogil'nitskiy and M.N. Shorokhov/

/Text/ For most methods, the existing practice of geophysical work presupposes the presence of two stages. The first is the collection of information under field conditions, and the second is the processing and interpretation of the obtained data in an office. The information collection stage is conducted primarily during the summer, while the processing stage takes place in the winter. This applies in equal degree to magnetic, gravitational, electrical and seismic surveying.

The large temporal gap between the collection of the data and the obtaining of the final results has a negative effect on the quality of surveying work and makes it impossible for geophysicists to approach purposefully the study of areas that are promising as far as mineral deposits are concerned. Besides this, the absence of automation in data collection, processing and interpretation leads to additional errors.

At the present time the most promising technique is working with areal observation systems that make it possible to study the spatial structure of physical fields that are being investigated. Especially attractive is the use of areal systems in exploration seismic surveying (the visualization of seismic mediums) /1/ and in seismology (to study temporal changes in the structure of the seismic wave field in seismically active zones).

The problems enumerated above can be solved by creating field computer complexes for collecting and processing geophysical information.

The Computer Center of the USSR Academy of Sciences' Siberian Department has begun working on the creation of such a complex, based on the "Elektronika NTs-OZT" microcomputer and the "CAMAC" main line-modular system.

The conditions for the conduct of geophysical experiments can be extremely variegated, and there are different methods for distinguishing signals and interpreting the information that is obtained. Consequently, a computer complex must be easy to reorganize in both the equipment and programming respects. The combination of the "CAMAC" system and the microcomputer provides just this flexibility.

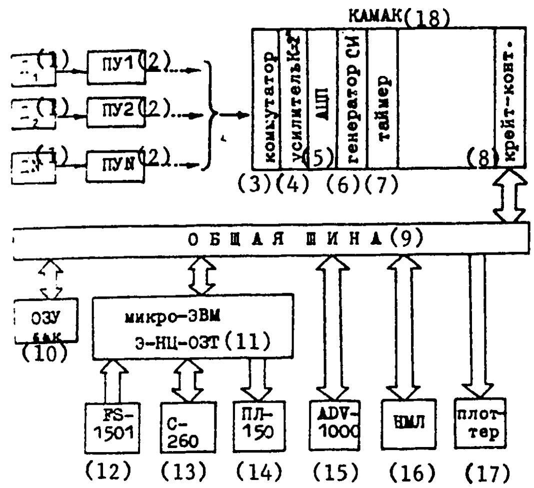


Figure 1.

Key:

1. Sensor
2. Peripheral unit
3. Commutator
4. K-2 amplifier
5. Alphanumeric printer
6. SI generator
7. Timer
8. Kreyt-controller
9. Common bus
10. 64K main memory
11. "E-NTs-OZT" microcomputer
12. FS-1501 photographic counter
13. Consul-260 electric typewriter
14. PL-150 keypunch machine
15. ADV-1000 display
16. Magnetic tape storage
17. Plotter
18. CAMAC

Figure 1 shows the layout of the complex. The set of modules represented in the CAMAC-kreyt /translation unknown/ makes it possible to provide the system's simplest function, which is the conversion of the sensors' analog signals into digital form over several channels. The large set of modules that is being produced at the present time makes it possible to expand the kreyt's functions with respect to the primary transformation of the incoming signals and their transmission into (for example) line communication networks. The order of the modules' functioning is assigned by the central processor, which is based on an "E-NTs-OZT" microcomputer. Having a sufficiently large memory (64K), as well as magnetic tape storage, the microcomputer makes it possible to realize quite complex algorithms for the operational processing of geophysical data.

In addition to the regular set of peripheral gear (FS-1501 photographic counter, "Consul-260" electric typewriter, PL-150 keypunch machine), the system, which has a common bus, contains a graph plotter, a display unit and a magnetic tape storage unit. The first two make it easier to interpret the results of the data processing. On the whole, the complex has the technical characteristics shown in Table 1. In order to increase the complex's productivity, a set of processors (up to 16) can be used in it. In connection with this, it is expected that the calculating speed will be increased by several hundred percent /2/.

Table 1. Technical Characteristics of the Complex

1. Analog information input:	
dynamic range of signals	150 dB
number of input channels	32
maximum input signal quantization frequency	10 kHz
2. Central processor ("Elektronika NTs-OZT"):	
word	16 bits
main memory volume	8K words, with expansion to 64K
number of commands	190
operating speed	50,000 operations/second of the addition type ("register-memory")
organization of communications	"common bus"
number of interrupt levels	8

Both the multiprocessor nature of the complex and the diversity of the possible processing algorithms is insured by the presence of the appropriate software, which includes the following systems.

A system of facilities for monitoring the functioning of the complex that makes it possible to establish the location and nature of a breakdown on the basis of the running of testing programs. When packages of applied programs are prepared under fixed conditions, a cross-system of programming on a BESM-6 high-speed computer is used; this makes it possible to obtain a controlled process program that is preliminarily written in the "BASS" special symbol-programming language. The cross-system has great possibilities for placing programs in archives, so the formulation of a package can be accomplished comparatively rapidly if only isolated new fragments are used in it. In this same system there is an interpreter and a debugger for the programs that are drawn up, which makes it possible to obtain finished controlled process programs directly on the big computer.

Changes can be made in a finished program when it is in operation in the microcomputer if debugging programs and a text editor are used. Under field conditions, a "BASS" resident assembler is used to create programs. "BASS" is a symbol-encoding language that has been written espeically for the microcomputer used in the complex. In the basic software there are also a punched-tape operating system and a set of program for expanded arithmetic and elementary functions. These programming systems are used as the basis for creating packages of applied programs, the primary ones of which are an applied statistical analysis package, a geophysical interpretation package, a seismic processing package and so on. The packages are preserved both on punched tape and on magnetic tape in small information carriers.

On the whole, as the applied processing software and the system's productivity are expanded and increased, it is possible to solve a broad spectrum of problems ranging from simple buffering of data to the construction of geophysical profiles.

BIBLIOGRAPHY

1. Alekseyev, A.S., et al., "Seismic Holography and Photography: Working Methods and Results," in "Problemy vibratsionnogo prosvechivaniya Zemli" /Problems in Vibration Radioscopy of the Earth/, Moscow, Izdatel'stvo Nauka, 1977, pp 32-53.

2. "Mul'tiprotsessornyye sistemy i parallel'nyye vychisleniya" /Multiprocessor Systems and Parallel Computations/, Moscow, Izdatel'stvo Mir, 1976, 256 pp.

/67-11746/

11746
CSO: 1863

UDC 551.1:550.34

ON-BOARD SYSTEM FOR THE PRIMARY PROCESSING OF GEOPHYSICAL DATA, BASED ON THE USE OF A GENERAL-PURPOSE COMPUTER

Novosibirsk PROBLEMY VIBROSEYSMICHESKIKH METODOV ISSLEDOVANIYA in Russian 1979 pp 52-62

/Article by V.N. Vizyayev, A.V. Gulyayev, V.N. Dement'yev and G.V. Larkin/

/Text/ The ideology of the formation of a Geophysical Information Processing Center that is being shaped at the VTs /Computer Center/ SO AN SSSR /Siberian Branch, USSR Academy of Sciences/ implies telemetric communication between a complex of large computers and remote specialized terminals that provide for the collection of experimental information and its preliminary processing. One of the forms of these terminals is an on-board measuring system for marine research.

VTs SO AN SSSR's Department of Applied Geophysics has now amassed a certain amount of experience in conducting marine research. In the experiments conducted on the hydrological research ship YaKOV SMIRNITsKIY in 1978-1979, the on-board measuring system was based on an LSI-2 minicomputer.

The basic goals of SO AN SSSR's first marine expedition included studying the microseismic background of heterotypical (in the geomorphic sense) structures in the Central Atlantic. The investigations of the microseisms involved the use of automatic, bottomset seismic stations and low-frequency hydrophones. In accordance with the problems that had been formulated, the on-board computer complex had to:

1. Provide for the deciphering, input and magnetic tape storage in the computer of the information arriving from the bottomset stations.

2. Provide for the digital recording (on a real time scale) and magnetic tape storage in the computer of information arriving from the hydrophones.

3. Provide preliminary processing of the stored data in order to monitor their quality.

4. Insure the development of the necessary applied programs.

In order to solve these problems, a computer complex was put together; a block diagram of this complex is shown in Figure 1.

The purpose of the individual elements in the complex requires no special explanation. Their basic technical characteristics are presented below.

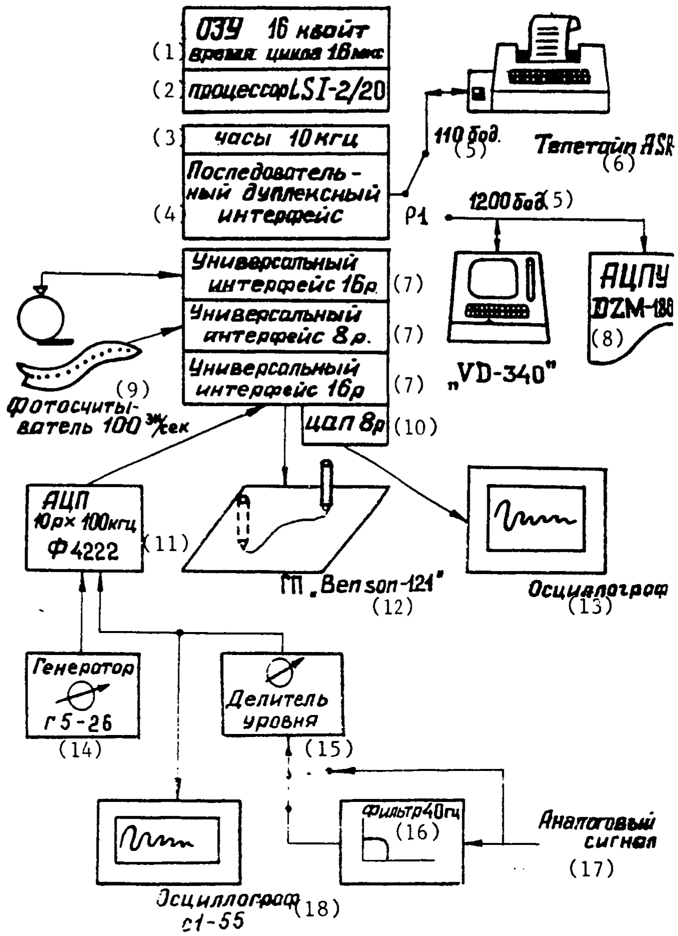


Figure 1. Configuration of on-board measuring system based on the LSI-2 minicomputer.

Key:

1. 16-Kbyte main memory (cycle time = 16 μs)
2. LSI-2.20 processor
3. 10-kHz clock
4. Sequential duplex interface
5. .... bauds
6. ASR teletype
7. ..-bit general-purpose interface
8. DZM-180 alphanumeric printer
9. Photographic reader (100 frames/s)
10. 8-bit digital-to-analog converter
11. F4222 analog-to-digital converter (10 bit x 100 kHz)
12. "Benson-121" graph plotter
13. Oscillograph
14. G5-26 generator
15. Level divider
16. 40-Hz filter
17. Analog signal
18. S1-55 oscillograph

Basic characteristics of the computer:
a) parallel processing of 16-bit words or 8-bit bytes;
b) four 16-bit general-purpose registers and one 8-bit computer-status register;
c) memory -- 8K 16-bit words; memory cycle -- 1.6 μs; capability for enlarging memory to 32K words per unit (capability for connecting up to 8 units with a total volume of 256K words);
d) capability for operating in direct-access-to-memory mode, in which case the maximum transmission speed is 940,000 words/s;
e) number of commands -- 188; average execution time -- 2 μs;
f) arithmetic operations performed on integers in two's complement;
g) has a large set of input/output commands, which makes it possible to organize operations with peripheral gear efficiently; number of physical numbers of units -- 256;
h) vector interrupt system;
i) number of sockets for connecting peripheral gear interfaces -- 14 (standard processor housing).

Basic characteristics of 0.5-inch magnetic tape storage unit, produced by (Keneli) Corporation (United States). Model 9000:
a) tape speed (working run) -- 37.5 in/s ($\sim$1 m/s);
b) recording method -- without reset to zero (NRZ 1);
c) recording density -- 800 bits/in ($\sim$32 bits/mm);
d) reading/recording performed in 8-bit bytes with a period of $\sim$33 μs;
e) recording format corresponds to the IBM (United States) standard.

The tape recorder control unit executes the following commands independently:
a) record "end of file" marker;
b) search ahead of "end of file" marker;
c) search after "end of file" marker;
d) space ahead of a single entry;
e) space after a single entry;
f) rewind tape;
g) erase 4 in of tape;
h) continuous entry; that is, entry of blocks without stopping between blocks. In this mode, the "space" between blocks is 500 bytes.

The maximum size of a block in the mode of automatic output by bytes is 64K bytes. In the programmed mode the size of a block is unlimited.

Quantization of an incoming analog signal is performed by an industrial F4222 10-bit analog-to-digital converter, the maximum quantization frequency of which is 100 kHz. The converter is set in operation by a G5-26 pulse generator.

Graphic information output is realized with a "Benson-121" step-type roller graph plotter, the minimum step of which is 0.1 mm; the stylus movement rate is 5.4 m/min (900 steps/s).

The use of a high-speed "Videoton-340" display and a DZM-180 alphanumeric printer made it possible to speed up the writing and debugging of information collection and processing programs considerably. The display was connected to the processor's standard teletype channel through switch P1. At the same time, the processor switched the transmission rate from 110 to 1,200 bauds. The DZM-180 alphanumeric

Table 1. Systems Modules of General LSI-2 Software

Name of Module	Functions of Module in System
MGR	Library of binary programs on magnetic tape. In the dialog mode, performs all the service functions of a librarian.
OMEGA	Dialog assembler. Makes it possible to edit original textual information and obtain control process modules of segments of a program.
LAMBDA	Editor of controlled process modules. Makes it possible to assemble a binary program, in the memory, from several individually prepared controlled process modules.
DEBUG	Makes it possible to edit and debug a binary program in the memory.
BDP	Makes it possible to discharge a binary program from the memory (onto PRL /possibly punched tape/).
BLD	Makes it possible to enter a binary program in the memory.
AUTOLOAD	Makes it possible to enter, from the computer's pulse /sic -- probably console/, binary loader BLD or library MGR in the memory.

printer was preliminarily equipped with a sequential reception interface and connected in parallel with the display's reception channel. Thus, when necessary it is possible to obtain a "hard" copy on the alphanumeric printer at the same time that information is being displayed on the screen.

During the 1978-1979 voyage, the LSI-2's general software (see Table 1) made it possible to produce programs for the input of single-channel analog information in real time in the frequency range up to 4 kHz, programs for the energy and spectral analysis of information stored on magnetic tape, and programs for the output of the original data and the results of preliminary processing, through the graph plotter, in the "ON LINE" mode.

The general structure of the system's special software is shown in Figure 2. It consists of a set of applied programs, each of which is based on the library of applied modules. This structural organization of the software made it possible to separate the work for the creation of applied programs rationally and to lay the foundation for developing a package of programs for the terminal.

The use of these programs on the voyage made it possible to investigate infra-acoustic noise on a real time scale, in sessions lasting up to 6 hours. The work was done with a hydrophone while the ship was drifting. All of the incoming information was stored in the computer on magnetic tape. In this mode, the recording unit's dynamic range was 57 dB.

When working with the bottomset stations, the programs that were written made it possible to calibrate the continuous recording channel and provided for deciphering of typical sections of the recording from microseisms.

The energy and spectral analysis programs made it possible to evaluate the quality of the incoming information during the voyage itself.

As examples, Figures 3-6 show the results of the operation of the complex of applied programs. Figures 3 and 4 represent realizations of microseisms recorded by a bottomset station and a hydrophone.

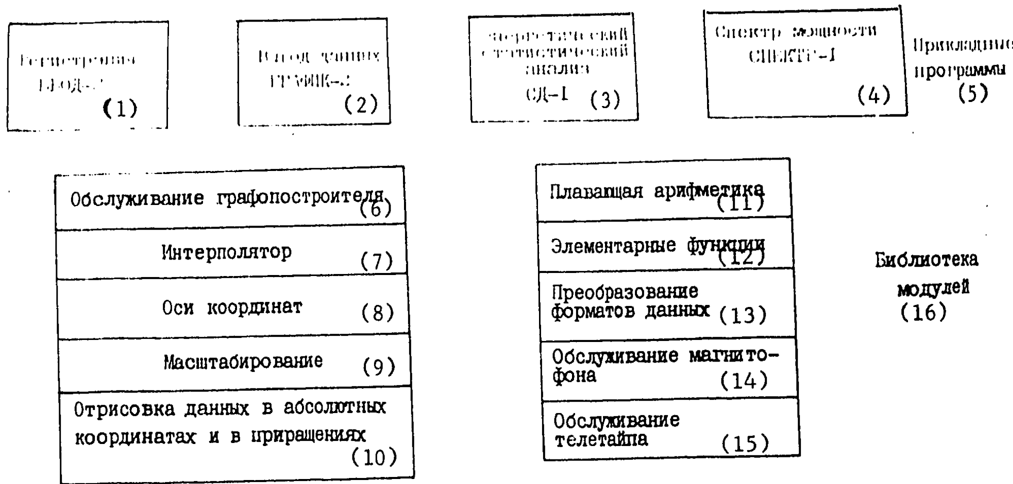


Figure 2. Structure of special software.

Key:

1. INPUT-2 registration
2. GRAPH-2 data output
3. SD-1 energy statistical analysis
4. SPECTRUM-1 power spectrum
5. Applied programs
6. Graph plotter servicing
7. Interpolator
8. Coordinate axes
9. Scaling
10. Depiction of data in absolute coordinates and in increments
11. Floating arithmetic
12. Elementary functions
13. Transformation of data formats
14. Tape recorder servicing
15. Teletype servicing
16. Library of modules

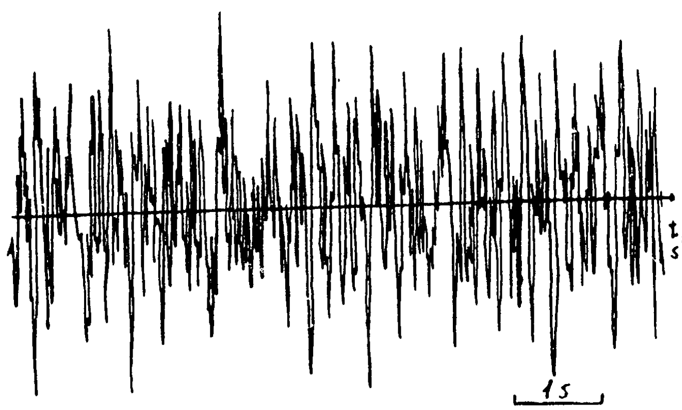


Figure 3. Example of recording of microseisms from a bottomset station.

Figure 5 represents a section of the energy analysis of a recording from a bottomset station, which makes it possible to visualize temporal changes in the level of microseisms (in the 2-80 Hz frequency range). Figure 6 shows a typical power spectrum from a bottomset station, on a linear level scale. On the recording it is possible to distinguish the harmonic interference from the motor of the bottomset

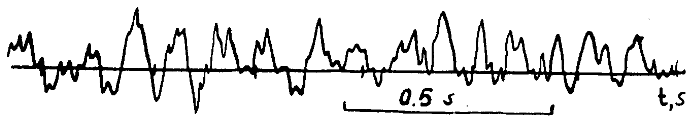


Figure 4. Example of recording of microseisms by a hydrophone.

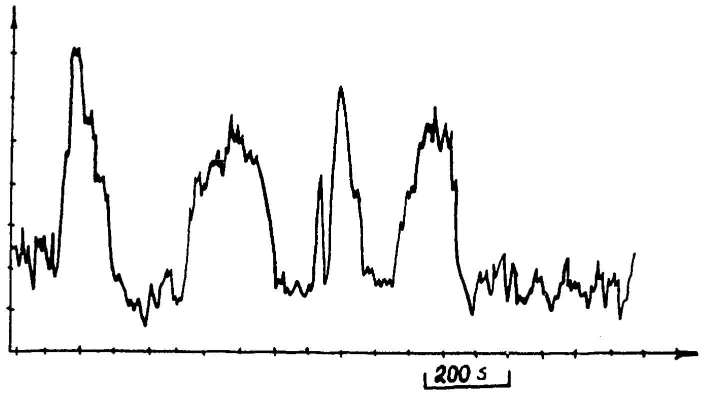


Figure 5. Temporal change in microseism power.

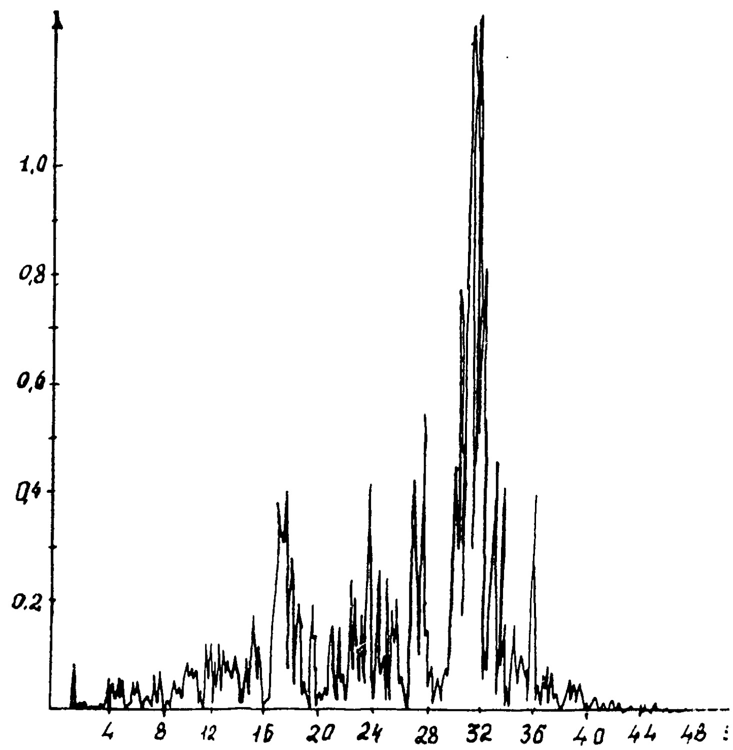


Figure 6. Power spectrum of bottomset station recording.

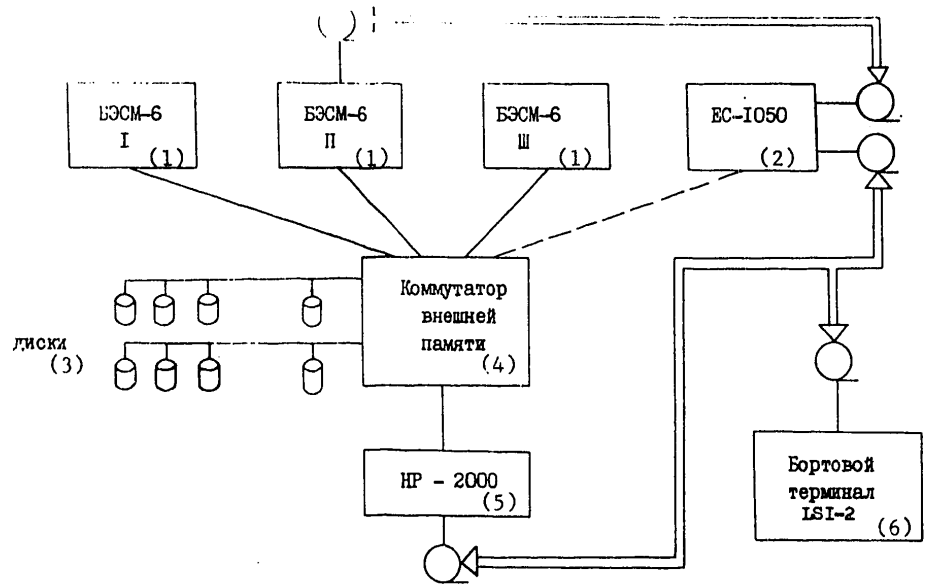


Figure 7. Diagram of on-board measuring system's communication with the complex of large computers at VTs SO AN SSSR's GP.

Key:

1. High-speed BESM-6 computer No ...
2. YeS-1050 computer
3. Disks
4. External memory commutator
5. NR-2000
6. LSI-2 on-board terminal

staton's tape-drive mechanism (at frequencies of 30-32 Hz). The absence of spectral components at frequencies above 60 Hz is evidence of the correct choice of the signal discretization frequency and its suitability for further processing. Mention should be made of the presence on the recording of the 2-Hz spectral line, which is well known in continental research and the origins of which have been discussed repeatedly by European seismologists (see /1-3/).

One characteristic feature of the on-board computer system that has been developed is the communicability of the information stored on magnetic tape with the complex of large computers at VTs SO AN SSSR's GP (see Figure 7). Essentially, the transmission channel for the information on the magnetic tapes can be regarded as a model of a telemetric communication channel. Putting this channel into operation makes it possible to test large data processing programs on real experimental material.

In conclusion, the authors wish to express their gratitude to Chief Engineer N.V. Kul'kov of VTs SO AN SSSR for his comprehensive collaboration in the development of the system and his suggestion that the LSI-2 computer be used in it.

BIBLIOGRAPHY

1. Pirhonen, S.E., and Luosto, U., "Spectral Observations on Artificial Seismic Noise in Southern Finland," PUBL. INST. GEOPHYS. POL. ACAD. SC., 1977A-5(116), pp 35-44.

2. Hjortenberg, E., "Monochromatic Components of the Seismic Noise in the 1-3 Hz Band," PUBL. INST. GEOPHYS. POL. ACAD. SC., 1977A-5(116), pp 21-23.

3. Plesinger, A., and Wielandt, E., "Seismic Noise at 2 Hz in Europe," J. GEOPHYS., Vol 40, 1974, pp 131-136.

/67-11746/

11746
CSO: 1863

UDC 551.1:550.34

EXPERIMENTAL RESEARCH IN THE DISCRIMINATION OF WEAK SEISMIC SIGNALS IN TERRESTRIAL VIBRATION RADIOSCOPY PROBLEMS, UTILIZING THE NR-2100 COMPUTER

Novosibirsk PROBLEMY VIBROSEYSMICHESKIKH METODOV ISSLEDOVANIYA in Russian 1979 pp 123-129

/Excerpts from article by A.S. Alekseyev, A.T. Gorbachev, B.V. Boshenyatov, B.M. Glinskiy, M.S. Khayretdinov, M.N. Shorokhov and V.M. Chupin/

[Excerpt] The system consists of two basic units: a stationary unit (located near the compressor) for transmitting the reference signal over the radio channel, and a movable unit mounted on a GAZ-66 truck.

The stationary unit contains: a seismic sensor (BEGIK); a filter tuned to the frequency 2.08 Hz; a reference signal shaper, which consists of multiplication circuits (50 Hz x 64) and a "tie in" to the seismic signal's phase, as well as a divider at the outlet of which a signal with a frequency of 2.08 Hz is formed. The reference signal is emitted with the help of a "Polosa-2M" radio transmitter; in connection with this, the signal is preliminarily modulated by a frequency of 2 kHz.

The movable unit contains: equipment for recording seismic signals with entry in an ASS-ZM ("Cherepakha") analog seismic station; an analog synchronous storage unit of the series type, with output of the results to a KSP-4 automatic recorder; a "Zond-I" digital storage unit. The movable unit also has a three-component SK-IP sensor and a narrow-band filter tuned to the frequency of the seismic signal emitted by the compressor. In addition to this, the movable unit contains a receiver to pick up signals from the "Polosa-2M" radio and a device for distinguishing synchronous impulses arriving over the radio channel.

The system makes it possible, under field conditions, not only to record seismic information, but also to process it in real time.

The subsequent processing of the recorded signals is carried out by a seismic signal input and processing complex consisting of an MT5528 magnetic recorder, a selective voltmeter with a variable frequency, a channel for the simultaneous input of four signals into an NR-2100 computer, and a plotter for depicting the results of the processing in graphic form.

The systems described were used to record the seismic field at different distances from an industrial source. The use of the field system made it possible to realize the algorithm for the noncoherent storage of signals on a real time scale.

The NR-2100 computer was used to realize algorithms for the cross-correlation method of distinguishing weak vibration signals against a background of interference, to conduct a spectral analysis of microseisms, and for the narrow-band, noncoherent storage of weak vibration signals and the spectral analysis of seismic signals with super resolution. The results of the investigations are described in several articles in this collection and in the preliminary report /2/.

BIBLIOGRAPHY

1. Alekseyev, A.S., et al., "Seismic Holography and Photography: Working Methods and Results," in "Problemy vibratsionnogo prosvechivaniya Zemli" /Problems in Vibration Radioscopy of the Earth/, Moscow, Izdatel'stvo Nauka, 1977, pp 32-53.

2. "Registratsiya i obrabotka vibroseysmicheskikh signalov i mikroseysm na osnove EVM (Otchet)" /Recording and Processing of Vibroseismic Signals and Microseisms, Based on the Use of a Computer (Report)/, Novosibirsk, Computer Center, Siberian Branch, USSR Academy of Sciences, 1979, 120 pp.

/67-11746/

11746
CSO: 1863

UDC 681.324

NETWORK ARCHITECTURE OF COLLECTIVE-USE COMPUTER CENTERS

Riga AVTOMATIKA I VYCHISLITEL'NAYA TEKHNIKA in Russian No 6, 1980 pp 3-13

[Article by E.A. Yakubaytis]

[Excerpts] Multi-console terminal systems are created most easily by using SM series minicomputers, and single-console by using microcomputers of the ELEKTRONIKA type. Together with these, for the purpose of implementing terminal systems can also be used minicomputers produced abroad, e.g., the YeS-1010, the Hewlett Packard 1000, the Wang 2200, etc.

At the Latvian SSR Academy of Sciences Institute of Electronics and Computer Technology has been created a communications system whose functioning satisfies the requirements of MKKTT [International Consultative Committee for Telegraphy and Telephony] Recommendation X.25. The center of this complex is an SM-3 or SM-4 minicomputer. To the common line of this minicomputer are connected microcomputers whose structure is illustrated in fig 8. The K 580 processor of the IK 80 microcomputer makes it possible to execute simple commands in 2.5 microseconds and operates with six levels of assignment priorities. The reprogrammable read-only memory and direct-access memory have a capacity of 64 K bytes. The carrying capacity of the line is 400 K bytes per second.

The microcomputer's bit-oriented controller makes it possible to link it with the BITKANAL [bit channel], and byte-oriented with the BAYTKANAL [byte channel]. To each of these controllers is connected a standard modem (48 K byte capacity) or a specially created data transmission unit making it possible to operate through a direct channel (two pairs of lines for a duplex or one pair for a semi-duplex) at speeds of up to 250 K bits per second. The common line controller makes it possible to link the microcomputer with an SM-3 or SM-4 minicomputer. The operator console's controller makes it possible to connect equipment required for debugging the microcomputer's programs. The latter are entered from a cassette tape recorder connected to the microcomputer by means of an appropriate controller, or from the memory of an SM-3, as well as an SM-4, minicomputer.

Bibliography

1. Yakubaytis, E.A. "Arkhitektura vychislitel'nykh setey" [Architecture of Computer Networks], Moscow, Statistika, 1980, 320 pages.

2. "Reference Model of Open Systems Interconnection (Version 4 as of June 1979)," ISO/TC 97/SC 16, 1979, N 227, 181 pages.

3. "CCITT Sixth Plenary Assembly, Geneva, 27 September - 8 October 1976, Orange Book, Vol VIII, 2, Public Data Networks," Geneva, 1977, 217 pages.

4. "DPS-25. Packet Switching System," (advertising prospectus of the French SESA firm), 1979, 20 pages.

8831
CSO: 1863

UDC 681.327.8

DATA TRANSMISSION MULTIPLEXORS

Moscow MUL'TIPLEKSORY PEREDACHI DANNYKH in Russian 1980 pp 2-5

[Annotation and foreword from book by A. I. Korchinskiy, A. F. Bulychev, A. R. Golubtsov, et al, V. S. Lapin (deceased) and A. I. Korchinskiy, editors, Energiya, 160 pages]

[Text] The basic principles of the organization and construction of data transmission multiplexors (MPD--multipleksor peredachi dannykh) are presented. Distinctive features of microprogrammed control and its realization in a multiplexor are examined. A description is given of existing multiplexors and the prospects of their development are reviewed. The place and role of multiplexors in ASU construction are shown.

The book is intended for engineering, technical and scientific workers engaged in the development of computer hardware and also instructors and students of the corresponding specialties.

Foreword

The task of further development and increase of the effectiveness of ASU and computer systems, the creation of the State Data Collection and Processing System for accounting, planning and management and the creation of shared multicomputer centers and data transmission and remote processing systems and networks and computer networks require with special urgency the assurance of the coupling of computer hardware with communication channels and networks, with organization of the access of distant users to computers. This advances the task of connecting computer centers within the limits of a single and different automated data processing systems.

The question of the creation and development of devices for coupling computers with communication channels became urgent when computers began to be used as the controlling link in data transmission and remote processing systems. Computers which included coupling devices appeared. Later, with the development of computer hardware, coupling devices were separated from the computer. However, such devices have not found wide application because of their specific functions and the impossibility of using them in other systems. There now are coupling units of a new generation--multiplexors for data transmission (MPD), which assure the creation of automated data processing systems.

The first Soviet series-produced multiplexors appeared as parts of YeS computer hardware. They included the MPD-1A, MPD-2 and MPD-3 multiplexors.

The development of computer coupling devices is now becoming an independent direction of development of computer hardware. Multiplexors, which obtained their start in simple coupling devices, are acquiring a new quality which consists in the expansion of their functions connected with the partial processing of transmitted information. Programmed MPD's and coupled processors are becoming more and more widespread in the creation of automated data processing systems.

The present book is devoted to the examination of the principles of construction of multiplexors and the clarification of problems in their development.

With consideration of the requirements of the mass user of YeS computers, a considerable portion of the material is devoted to YeS computer multiplexors. Material on the model YeS-8402 programmable MPD also is widely represented in the book.

The book consists of three chapters.

Chapter 1 contains an analysis of the principal functions and distinctive features of the use of MPD's, their role and place in the ASU, the influence of information flows on MPD selection. The prospects of the development and use of MPD's are shown. Coupling processors are examined as a new direction in coupling development. The material of this chapter reveals the problems which must be solved in an ASU in selecting coupling devices.

Chapter 2 is devoted to consideration of methods of organizing MPD construction and its coupling with a computer and subscribers. Problems in the organization of methods of access and protection during data transmission and the organization of priorities are discussed and approximate recommentations are developed.

Chapter 3 presents the basic principles of MPD construction and a general classification of them. The principles of synchronous and asynchronous control in MPD's are examined. Distinctive features of MPD's with circuit and microprogram control are shown. The material of the chapter can help the reader clarify distinctive and specific features of the principles of MPD construction.

Chapter 4 contains a brief survey of the Soviet MPD-1A, MPD-2 and MPD-3 multiplexors. The principles of the apparatus realization of MPD's are examined in the chapter, and an algorith for their functioning is presented. Distinctive features of the application of the MPD-1A, MPD-2 and MPD-3 are shown.

In Chapter 5 the principles of the organization and construction of a programmed MPD are examined, its architecture is given and distinctive features of the control unit of a programmed MPD are reviewed.

Specialists in computer technology, communications technology, data transmission and the planning of data transmission systems, including ASU and shared multicomputer centers, will find necessary practical material in the book. In writing the book the authors also took into consideration users who work with remote data processing and transmission equipment.

The introduction, Chapter 1 (except section 1-4) and Chapter 2, section 4-1 were written by A. F. Bulychev, Chapter 3 by A. I. Korchinskiy and A. F. Bulychev, Chapter 4 (except sections 4-1 and 4-7) by V. I. Utkin and V. S. Lapin (deceased), sections 1-4 and 4-7 by A. I. Filatov, and Chapter 5 by A. I. Korchinskiy, A. F. Bulychev and A. P. Golubtsov.

[54-2174]

2174
CSO: 1863

UDC 62-52:681.3.06.44

FORMS OF DATA REPRESENTATION ON COLOR VIDEO TERMINAL SCREENS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980 pp 28-29

[Article by N. V. Rzhenkovskiy, senior scientific associate, Zaporozh'ye branch, All-Union Scientific Research and Design Institute "Tsvetmetavtomatika"]

[Text] Video terminals with color screens are now in a state of intensive development and introduction in automated dispatcher control systems. They assure very complete contact of man and computer thanks to the varied transmitted information (color, graphic and textual). Such contact is necessary in complex control systems working in real time to increase the effectiveness of man-computer interaction and reduce the probability of erroneous actions of the system operator [1].

The possibilities of color graphic displays and various forms of data representation on them are fairly well known [2]. However, in practice work with a display is tiresome for man because of the large volume of visual information. Therefore the main factor in the creation of new forms of data representation are their configuration and the clearness and field of vision of the representation.

Proposed as one of the ways to achieve that goal is the formation of an image on a screen in the form of mnemonic diagrams, which permit combining the mnemonic diagram of a technological process with the necessary parameters of the object of control. A mnemonic diagram represents a stylized image of controlled equipment grouped according to certain functional criteria, to which the graphic representation of the regimes and parameters of the technological process have been reduced in position.

On page 3 of the booklet, at the top, is an example of a summary mnemonic diagram of technological processes of the electrolysis of metal (the values of the parameters are presented provisionally). The form of the basic elements of the given mnemonic diagram represents a horizontal line-- a base which provisionally designates a group of technological equipment on which deviations of the controlled parameters from the nominal values (upward from the base) and the working conditions of the equipment (downward from the base) are depicted by vertical vectors. The vectors are localized in definite positions of the base corresponding to specific units of the technological equipment.

On each such unit of equipment, as a rule, a group of parameters is monitored, but to preserve the factor of visibility only one factor of the group which has a maximum deviation from the norm is indicated.

The indication is done in relative units of the range of regulatory values, limited to elements of the mnemonic diagram by a horizontal segment set above the base.

The basic color of indication is green. In the case of deviation of a parameter from the norm the portion of the vector characterizing emergency values is indicated withred light. From the moment an emergency situation appears a glimmer of the red part of the vector-parameter occurs before acknowledgment of receipt of information by the operator.

Vector regimes display color information about certain states of the technological equipment in the time scale. On the cited mnemonic diagram the green color designates the metal selection regime, blue color the correction of the electrolyte composition and white the disconnected electrolyzer.

To reveal generalized information of the survey mnemonic diagram, fragment mnemonic diagrams with a different degree of detailing are used. An example is the mnemonic diagram of a group of electrolyzers on which information about one group of the technological equipment is given in detail (see page 3 of the booklet, photo in the lower left).

Presented on the given mnemonic diagram are the numbers of the electrolyzers, the scale dimensions, the figures of parameters with emergency values, conventional symbols and some additional information about the technological process.

Besides problems in representation of information about the state of a technological object of control, in automated systems based on computer hardware a need arises for monitoring the functioning of a hardware complex. Such monitoring can be done by means of the mnemonic diagram presented in the same place, on page 3 of the booklet, the photo in the lower right. On that mnemonic diagram the complex of the automated system for control of technological processes of an electrolyzer is depcited as an example. The information of the mnemonic diagram is divided into two parts corresponding to shop sections.

Mnemonic diagram elements are the markers designating by position the monitored devices, with color information about their working conditions:

yellow -- the state of the "warm reserve"
white -- disconnected equipment
blue -- the work of equipment with defects
red -- inoperative equipment
green -- normal functioning

For the purpose of information ordering on a screen the indication of all equipment of the same type for a measured parameter (temperature, for example) is made in the form of a two-dimensional array of markers.

Side by side with giving details of information, a mnemonic diagram also contains survey information (outlined sections on the right side of the screen) on the functioning of sensors and corresponding hardware. The survey information is also arranged in the form of similar sections on technological mnemonic diagrams and represents an integrated estimate of the working capacity of monitored equipment. Each element of the survey sections characterizes a group of similar devices in a cross-section of the measured parameters. The devices can have different states and therefore the indication of group markers is done with the following priority:

1 -- disconnected equipment
2 -- inoperative equipment
3 -- work with failures
4 -- normal work

The presence of survey information on the functioning of a hardware complex on technological mnemonic diagrams permits rapid visual monitoring of the reliability of the values of parameters of the objects of control, which is especially important in emergency situations.

The described method of information representation is the basis of the automated system for the control of technological processes developed by the Zaporozh'ye branch of the All-Union Scientific Research and Design Institute "Tsvetmetavtomatika" for the electrolysis shop of one of the enterprises of the sector. The automated system is constructed on the basis of an SM computer with an A543-11 color graphic terminal [3].

The application of the proposed forms of representation in the form of display on video terminals in the form of color mnemonic diagrams is advisable above all in automated systems for dispatcher control of technological processes. It will permit the user of a system to considerably reduce the time required for analysis of information in comparison with tabular and textual forms of its representation, and also increasing the quality of interaction with the computer.

BIBLIOGRAPHY

1. Cherednichenko, N. F., Tkachenko, G. D., et al. Device for display of dispatcher information on a color cathode-ray tube. USiM, 1978, No 5, pp 129-132.

2. Soloveychik, I. Ye. Displei v sistemakh s EVM (Displays in Systems with Computers). Sovetskoye radio, 1979, 248 pages.

3. Tsvetnoy graficheskiy terminal A 543-11. Arkhitekturnoye opisaniye (The A 543-11 Color Graphic Terminal. Architectural Description). Severodovetsk, NIIUVM, 1979, 26 pages.

[54-2174]

2174
CSO: 1863

AUTOMATED DESIGN FOR METAL CONSTRUCTION ELEMENTS BRINGS SAVINGS

Moscow VESTNIK AKADEMII NAUK SSSR in Russian No 10, 1980 pp 42-53

[Article by Academician N. P. Mel'nikov and Doctor of Technical Sciences G. A. Gemmerling: "System for Automating the Design of Steel Construction Elements"]

[Excerpts] Intensive capital construction in the USSR entails a constant increase in the use of steel construction elements. A significant share of them are manufactured on the basis of designs developed at Soyuzmetallostroyniiproyekt [possibly USSR Scientific Research and Planning Institute of Use of Metal in Construction] and its head institute, TsNIIPSK [Central Scientific Research and Planning Institute of Steel Construction Elements]. Work in the area of improving the efficiency of metal use in construction pursues both improvements in the grades of steel and assortment of rolled products used in construction and fundamentally new, progressive design forms for buildings and structures. One of the most important reserves for increasing the efficiency of metal use in construction is automating the design of construction projects. TsNIIPSK is working on this in contact with several leading academy and sectorial institutes in the country whose specialists are represented in the "Construction" section of the Scientific Council on the Problem "Development of Systems Methodology for Solving Problems of Automating Design," which is part of the Committee on Systems Analysis of the Academy of Sciences USSR. The present article is devoted to an analysis of the problems of formulating a system for automated design (SAPR) of construction elements and to the first results of using such a system in design work at TsNIIPSK.

In the TsNIIPSK SAPR all the necessary components are developed for automated work positions based on a set of YeS-7906 alphanumeric displays that make it possible to solve design problems efficiently with a small volume of initial and resulting information and low calculation time. Needless to say, the automated work position can only be used by specialists working a short distance from the computer. Other data processing schemes must be worked out for remote users.

The Economic Impact of Introduction of the SAPR

Development of an SAPR of the volume and complexity of the TsNIIPSK SAPR requires 4-5 years of labor by a collective of about 100 systems developers and capital investments of 3-4 million rubles. Therefore, the question of economic substantiation of the wisdom of developing an SAPR is very critical.

The economic impact from introduction of an SAPR results from two factors. Above all the quality of planning decisions is improved, which reduces the expenditure of steel for construction elements and improves the reliability of the structure and its operating characteristics. Optimization of the parameters of a structure usually leads to a direct savings of steel. The usual target function selected is the weight of the construction element, but other indicators of the planning decision (deformation characteristics, coefficient of safety against buckling, and the like) are put in the optimization model as constraints. This is how the problem of optimization of towers is formulated in the TLP [technological design line] for towers; when it is used the weight of the elements designed is roughly 10 percent less than similar designs developed within the traditional system of design work.

Subsystems for strength calculations of metal construction elements are used in practice much more broadly than subsystems for full automation of design work. The use of refined calculation schemes (rejection of excessive schematization of particular elements in the calculation scheme and using nonlinear models of the work of the material and calculation schemes that describe the work of the structure as a whole, considering a large number of possible combinations of loads, and the like) makes it possible to receive by automated calculation more reliable values of the extreme forces that arise in construction elements during their use. This makes it possible for designers selecting the cross-sections of elements to avoid creating unnecessarily high reserves of carrying capacity and leads to a savings of steel. It is thus possible to create construction elements in which the indicators of working reliability of particular elements are closer, which results in more reliable work by the structure as a whole. This leads to a savings of steel of up to five percent of the weight of the basic carrying elements of the project being designed (up to 2.5 percent of the full weight of the metal construction elements). The estimated reduction in calculated expenditures by improving the quality of design decisions when the first phase of the TsNIIPSK SAPR is fully introduced is 9.6-13.2 million rubles a year.

Another source of economic impact with the introduction of an SAPR is an increase in labor productivity in design work.

As already mentioned, labor productivity in the design part of TsNIIPSK rose at an average rate of 6.7 percent a year in 1971-1977, and roughly half of this growth resulted from the introduction of automated design systems. At the same time, the time required to perform planning jobs was greatly reduced (in some cases to one-half or one-third). This was especially notable in the cases of designing such complex projects as the convertor shops of the West Siberian and Novolipetskiy metallurgical plants and structures at the Kama Truck Plant.

The growth in labor productivity is most remarkable with introduction of SAPR subsystems of the tower-type TLP. This made it possible to begin switching the entire system for designing television towers in our country to a new basis. The design process for such towers was formerly divided into three stages: development of standardized design decisions for particular elements; formulation of model designs of structures composed of model and standardized elements; reconciling the model design of the structure with the specific construction site. Direct labor expenditures by designers in the development of each individual tower design using model and standardized elements was cut by more than two-thirds through the introduction of the TLP and became comparable to labor expenditures for reconciling the model tower design. This made it possible to eliminate the stage of development of model tower designs and to replace the stage of reconciling model designs with individual tower design work.

[60-11176]

11,176
CSO:1863

COMPUTER NETWORK AT INSTITUTE OF ATMOSPHERIC OPTICS

Tomsk TEKHNICHESKIYE SREDSTVA VYCHISLITEL'NOY SETI INSTITUTA OPTIKI ATMOSFERY in Russian 1979 pp 2, 79

[Annotation, table of contents of book "Technical Equipment of the Computer System of the Institute of Atmospheric Optics", Siberian Department of the USSR Academy of Sciences, 1979, 80 pages]

[Excerpts] This collection reflects the results of work to devise technical equipment for the research automation system of the Institute of Atmospheric Optics. The system comprises a central computing complex based on an ASVT-4030, a subsystem for communications within the system, and subsystems to automate physical experimenta.

The book has two parts. The first part contains articles which describe the principles of organization and technical equipment of the communications system which provides information exchange among the computers by cable and telephone lines. The second part describes devices for tying peripheral units into the computers and the main exchange system.

The material in this collection is oriented to specialists in the development of automated systems and to scientific associates who use the computing system.

Table of Contents

[61-11176]

11,176
CSO: 1863

GROUP ADAPTER FOR COUPLING BASE COMPUTER AND PERIPHERAL UNITS DESCRIBED

Tomsk TEKHNICHESKIYE SREDSTVA VYCHISLITEL'NOY SETI INSTITUTA OPTIKI ATMOSFERY in Russian 1979 p 5

[Article by G. F. Vakurov and V. I. Shishlov: "The Group Adapter of the Computing System of the Institute of Atmospheric Optics"]

[Excerpt] One of the promising lines of development in computer use for scientific research is building multimachine measurement and computing systems [1-3]. The questions of interlinking different types of computers and organizing automatic program-controlled information exchange among subsystems at different levels are very important in the development of multilevel systems for automating scientific research.

A group adapter has been developed for the computing system to automate scientific research at the Institute of Atmospheric Optics of the Siberian Department of the Academy of Sciences USSR [4]. The adapter provides interaction between a channel of the base M-4030 computer and nonstandard terminals, namely small MIR-2 computers, Videoton-340 displays, minicomputers, and program-controlled complexes of systems for automating experiments.

The adapter provides data input and output from the M-4030 computer on eight active nonstandard peripheral units by cable with a speed of up to 100 kilobytes per second. The terminals may be up to 100 meters away from the group adapter.

[61-11176]

11,176
CSO: 1863

INTERFACE WITH MAIN MEMORY MODULE DESCRIBED

Tomsk TEKHNICHESKIYE SREDSTVA VYCHISLETEL'NOY SETI INSTITUTA OPTIKI ATMOSFERY in Russian 1979 p 53

[Article by V. I. Naumov: "The Interface of the Main Memory Module"]

[Excerpt] The interface is designed to organize data exchange between the main KAMAK line and the main memory module at the Leningrad Svetlana Association of Electronic Instrument Making and for autonomous work by the main memory module with an analog-digital converter.

When used with the main line the interface can read or write 2,048 16-byte words or write 2,048 or 4,096 eight-byte words working in an autonomous regime; in both cases the access cycle of the main memory module interface is four microseconds. It is possible to connect two interfaces to one alphanumeric converter to increase the data writing speed and memory volume.

In the autonomous mode the work of the main memory module is controlled through supplementary sockets of the interface and alphanumeric convertor using two algorithms:

1. reading an eight-byte word from the alphanumeric convertor into the 16-byte shift register and writing it in the main memory module;

2. forming a 16-byte word from two eight-byte words and then writing in the main memory module.

[63-11176]

11,176
CSO: 1863

DRIVER OF M-32/RV-7 MAGNETIC DRUM STORAGE DESCRIBED

Tomsk TEKHNICHESKIYE SREDSTVA VYCHISLETEL'NOY SETI INSTITUTA OPTIKI ATMOSFERY in Russian 1979 p 43

[Article by Ye. Pokrovskiy and V. I. Potapov: "The Driver of the M-32/RV-7 Magnetic Drum Store in the KAMAK Standard"]

[Excerpt] The M-32/RV-7 magnetic drum store (capacity of 983,040 bytes) is an external storage unit of the Minsk-32 computer and is planned to work together with a UUMB [magnetic drum control unit] storage control unit. The storage driver couples the magnetic drum store with the main line of the KAMAK "kreyt" [translation unknown]. The module makes it possible to read and write in periods of 30 microseconds or 270 microseconds and automatically scans addresses of the magnetic drum store within the limits of one zone.

[63-11176]

11,176
CSO: 1863

PUBLICATIONS

UDC 551.465.11

MATHEMATICAL MODELS OF CIRCULATION IN THE OCEAN

Novosibirsk MATEMATICHESKIYE MODELI TSIRKULYATSII V OKEANE in Russian 1980 signed to press 19 May 80 pp 2-5

[Annotation and foreword from the book edited by G.I. Marchuk, academician of the USSR Academy of Sciences, and Doctor of Mathematical Physics A.S. Sarkisyan, Izdatel'stvo "Nauka", Siberian Division, 1,250 copies, 288 pages]

[Text] The monograph is devoted to the theory and methods of mathematical modeling oceanic circulating currents, which is being developed in Computer Center of the Siberian Department of the USSR Academy fo Sciences. A number of new formulations of problems of ocean dynamics are treated, their solvability is studied and efficient solution algorithms are proposed. Considerable attention is devoted to difference schemes and methods of solving the formulated problems. The results of calculations of the hydrothermodynamic fields are given and analyzed for individual seas and oceans as well as the world ocean as a whole.

The work is of interest to specialists working in the field of computer mathematics, hydrodynamics, oceanology and meterology.

The authors: Guriy Ivanovich Marchuk, Vladimir Pavlovich Kochergin, Artem Sarkisovich Sarkisyan, Mikhail Alekseyevich Bubnov, Vladimir Borisovich Zalesnyy, Viktor Ivanovich Klimok, Avtandil Aleksandrovich Kordzadze, Viktor Ivanovich Kuzin, Aleksandr Vasil'yevich Protasov, Vladimir Alekseyevich Sukhorukov, Yelena Aleksandrovna Tsvetova, Aleksandr Valentinovich Shcherbakov.

Foreword

The number of papers devoted to mathematical modeling of the thermohydrodynamics of the world ocean and its individual regions has grown rapidly in recent years. While numerical experiments on the dynamics of currents were a rare phenomenon (Sarkisyan, 1954, 1961, 1962) from the middle of the 1950's to the beginning of the 1960's, now the majority of theoretical studies of this problem are performed specifically using numerical methods. This has acted to assist the rapid development of computer mathematics and computer equipment.

A significant number of works have been published at the present time which are based on the utilization of nonlinear mathematical models realized on powerful computers. Some such models were constructed in Princeton (G.F.D.L.) by K. Bryan and his coauthors. The calculations of Bryan and Lewis (1979) as well as Manabe and Bryan (1979) are quite interesting. The work of Mintz, Takano and Han (1973) also deal with this area of research. However, studies have also ascertained substantial drawbacks to the mathematical formulations and algorithms for the numerical realization. Numbered among such drawbacks are exaggerated values of the coefficients of horizontal turbulent exchange, idealization and the formulation of the boundary conditions, etc., and for this reason, the problem of developing models for the overall circulation of the atmosphere and the ocean remains, as before, an urgent one. Special studies were performed in the Computer Center of the Siberian Department of the USSR Academy of Sciences in cooperation with the Institute of Oceanology of the USSR Academy of Sciences, which were devoted to a study of the global circulation of the world ocean, based on rather precise finite difference schemes and efficient algorithms. New algorithms were developed which provide for correct numerical models with physically real coefficients of turbulent exchange. Special schemes for improving the precision of the solutions were used, which were obtained using the method of embedded grids. An important step in the numerical solutions of global problems of oceanic circulations was the realization of numerical algorithms for the solution of systems of difference equations obtained by means of the method of finite elements. As is well known (Marchuk, 1977), the method of finite elements guarantees the preservation of determinacy of the original operator of the problems, and for this reason, the approximation of differential equations by difference equations, in this case, does not in principle distort the qualitative features of the solution of the original problem. It is likewise important to make the equations for turbulence in the active layer of the ocean more precise when modeling certain phenomena of oceanic circulation.

Along with equations of hydrodynamics, the Obukhov-Kolmogorov-Monin model, the turbulence energy equation and the equation for the rate of dissipation of turbulent energy in the ocean are also treated in the monograph. Such a mathematical model has made it possible to create a theory for the seasonal course of oceanic circulation, as well as ascertain the response of the ocean to storm processes and variations in the thermocline which are related to this phenomenon.

We will note that numerical modeling of oceanic circulation necessitated the performance of a broad set of studies of the correctness of the formulations of the problems. As a result, a broad class of correct formulations and their difference approximations was ascertained. The studies which were performed prepared a solid basis for the conduct of widescale numerical experiments with complex models of ocean dynamics, which permit a physical interpretation of the results.

Efficient numerical methods of solving world ocean thermohydrodynamics problems and the results of specific applications of these methods are also given in the monograph.

A description of the work done by Bryan, Manabe and Lewis, which was graciously made available by the authors, is included in the monograph (§ 6.1). These are their latest accomplishments in the field of numerical modeling of global oceanic circulations.

A number of mathematical formulations of problems of modeling the circulation of world ocean waters are treated in Chapter 1, which was written by G.I. Marchuk, A.A. Kordzadze and M.A. Bubnov, the correctness of these formulations if studied, as well as the existence and uniqueness of the solutions, and apriori estimates are given which are useful in the design of stable numerical schemes. Chapter 2 (authors G.I. Marchuk, V.P. Kochergin and V.A. Sukhorukov) is devoted to the mathematical modeling of the processes of vertical turbulent exchange and the mechanism for the formation of the seasonal thermocline in the ocean. Chapter 3 (V.B. Zalesnyy, A.A. Kordzadze, V.I. Kuzin, M.A. Bubnov) contains new difference methods for the solution of problems of the circulation of a baroclinic ocean. In Chapter 4 (V.P. Kochergin and A.V. Shcherbakov), economic difference schemes of a second order precision are constructed on the basis of the method of directional differences. Spectral methods of solving problems of ocean dynamics are treated in chapter 5, written by G.I. Marchuk and A.V. Protasov. These methods are specifically applied to the study of the dynamics of the tides of the barotropic world ocean and for the solution of the linearized problem of the dynamics of periodic currents of a baroclinic ocean. Chapter 6 is devoted to the application of the numerical methods set forth in Chapters 3 and 4, and to the study of the circulation of the baroclinic world ocean. There is a brief review written by A.S. Sarkisyan on the new achievements of foreign scientists working on this problem in § 1 of this chapter. The results of calculations of the currents and temperature fields performed by V.B. Zalesnyy are contained in § 2. A method of establishing a steady-state for a difference scheme is employed, which is given in chapter 3. In § 3, V.I. Kuzin treats the application of the method of finite elements to the solution of that same problem. In § 4, V.P. Kochergin and A.V. Shcherbakov analyzed the results of modeling the currents and density field of the world ocean, obtained by the methods given in chapter 4. Chapter 7 is devoted to local problems of the dynamics of seas and oceans. In § 1 and 2 of this chapter, V.A. Sukhorukov and V.I. Klimok study the seasonal variability of the currents of the North Atlantic based on the upper homogeneous model presented in chapter 2. In § 3, V.P. Kochergin and A.V. Shcherbakov solve the third boundary problem and give the results of calculating the density field and currents of the same basin. In § 4, A.S. Sarkisyan and D.G. Rzheplinskiy study the mechanism of the formation of the density fields and currents using the example of the North Atlantic (Neuman's problem is solved). A numerical model for the dynamics and thermal mode of Lake Baykal is presented in § 5, written by Ye.A. Tsvetova.

The authors would like to express their sincere gratitude to A.A. Konyakhinaya for the considerable assistance rendered in putting the manuscript together.

[62-8225]

8225
CSO: 1863

MATHEMATICAL METHODS OF INTERPRETING GEOPHYSICAL OBSERVATIONS

Novosibirsk MATEMATICHESKIYE METODY INTERPRETATSII GEOFIZICHESKIKH NABLYUDENIY in Russian 1979 signed to press 26 Dec 79 pp 4, 175-177

[Foreword and abstracts of papers in the collection edited by A.S. Alekseyev, Computer Center of the Siberian Department, USSR Academy of Sciences, 600 copies, 177 pages]

[Text] Foreword

This collection is devoted to questions of the application of mathematical computational methods in the interpretation of geophysical fields. The majority of the work published in the collection was performed in the department of mathematical problems of geophysics of the Computer Center of the Siberian Department of the USSR Academy of Sciences.

The papers of V.V. Voronin, V.N. Martynov and B.G. Mikhaylenko, and A.T. Fat'yanov and B.G. Mikhaylenko are devoted to numerical methods of calculating wave fields in elastic media. Inverse problems of wave propagation and diffraction theory, as well as photogrammetry are treated in the paper of V.D. Yelinov and G.N. Yerokhin. The papers of Sharafutdinov, V.A., Belonosova, A.V. and Tsetsokho, A.A. are devoted to questions related to the ray technique of calculating wave fields. The paper of A.G. Marchuk is devoted to questions of the generation and propagation of tsunami waves.

Abstracts of Papers Found in the Collection

UDC 550.344

THE CALCULATION OF GEOMETRIC DIVERGENCE IN CARTESIAN COORDINATES

[Belonosova, A.V., Tsetsokho, V.A.]

[Text] A system of six ordinary differential equations is derived in the paper to determine the geometric divergence of the central field of seismic rays in a medium without faults. In contrast to the works of other authors, only a cartesian system of coordinates is used in the derivation of the equations, and the characteristics of the medium are specified in this system. It is also demonstrated in the paper that the advantage of a particular system of differential equations for the numerical determination of geometric divergence is related not so much to the number of equations in the system as to the number of arithmetic operations needed for the calculation of the right side of the system at one point.

UDC 517.948:519.6

THE NUMERICAL SOLUTION OF THE TWO DIMENSIONAL PROBLEM OF ELASTIC WAVE DIFFRACTION AT AN ELASTIC BODY

[Varonin, V.V.]

[Text] The major features of an algorithm for the numerical realization of the method proposed and substantiated in previous literature by the author are described in this paper. The elimination of singularities is utilized to calculate the coefficients of the matrix derived following the digitization of a system of singular integral equations. The results of numerical experiments are presented. Figures 5; bibliographic citations, 4.

UDC 617.544

THE RESTORATION OF COEFFICIENTS IN THE CASE OF THE LOWEST DERIVATIVES IN AN ACOUSTICS EQUATION

[Yelinov, V.D.]

[Text] The question of the uniqueness of the determination of coefficients for the lowest derivatives in an acoustics equation is treated in this paper. The original problem is reduced to a well-known problem of integral geometry. Bibliographic citations: 3.

UDC 517.945

ON THE QUESTION OF THE STABILITY OF THE DEFINITION OF RADIATING POINT OBJECTS

[Yerokhin, G.N.]

[Text] The stability of the solution of the inverse problem of the determination of the "brightness" and coordinates of a finite number of shining sources with respect to information, which is specified in the form of some blurred image, is studied. A function characterizing this stability is determined in explicit form in two special cases.

UDC 550.345

METHODS OF CALCULATING TSUNAMI WAVES WITHIN THE FRAMEWORK OF APPROXIMATE MODELS

[Marchuk, A.G.]

[Text] A number of methods are proposed in this paper for the calculation of problems of the generation, propagation and departure of tsunami waves at a shore. The generation problem is solved within the framework of a non-linear system of equations for shallow water. It is also proposed that the model of an ideal incompressible fluid be used for the solution of generation problems. The characteristics of tsunami waves are studied by means of numerical experiments as a function of the motions of the bottom. To calculate problems of tsunami wave propagation in a basin of variable depth, a method is constructed for separate calculations in the deep and shallow portions of the basin. The question of tsunami wave energy is discussed in the paper, as well as its relationship to the energy of the focus. A more precise formula is proposed for the estimation of this energy. An original method involving a transition to an obliquely angular system of coordinates is proposed for the calculation of problems concerning the rolling-out of tsunami waves on a sloping shore. The results of numerical calculations using the indicated procedures are given. References 9; figures 22.

UDC 518.61.550.344

NUMERICAL MODELING OF ELASTIC WAVE PROPAGATION IN ANISOTROPIC INHOMOGENEOUS MEDIA (THE CASE OF A HALF-SPACE AND A SPHERE)

[Martynov, V.N. and Mikhaylenko, B.G.]

[Text] Algorithms are presented in the paper which utilize finite integral transforms in conjunction with numerical methods to solve problems of the propagation of elastic waves in a transversally isotropic half-space. The method is also extended to a radially inhomogeneous, tangentially isotropic model of the earth. The question of the application of finite integral transforms with respect to time to the solution of the problem of elastic wave propagation in inhomogeneous anisotropic media with absorption is also treated. References: 10.

UDC 518.61.550.344

A NUMERICAL SOLUTION OF LAMB'S PROBLEM FOR AN INHOMOGENEOUS BOLTZMANN MEDIUM WITH AN ELASTIC AFTER-EFFECT

[Fat'yenov, A.G., Mikhaylenko, B.G.]

[Text] The paper consists of two sections. Lamb's problem is solved in the first for an inhomogeneous (with respect to depth) half-space, filled with a Boltzmann medium with an exponential after-effect function. An approximation method is proposed in the second for the solution of the given problem for an arbitrary after-effect function. The convergence of the methods for a homogeneous medium is proved. Theoretical seismograms are given for a layer in the half-space with differing absorptions. References: 22; figures: 8.

UDC 513.73

ON GEOMETRIC DIVERGENCE

[Sharafutdinov, V.A.]

[Text] It is shown in the paper that the calculation of the geometric divergence can be reduced to the solution of Caushy's problem for Jacobi's equations, and the symmetry of geometric divergence in the case of an anisotropic medium is likewise demonstrated. References: 4.

[58-8225]

8225
CSO: 1863

UDC 681.32

PLANNING OF PARALLEL COMPUTATIONAL PROCESSES

Moscow PLANIROVANIYE PARALLEL'NYKH VYCHISLITEL'NYKH PROTSESSOV in Russian signed to press 28 Apr 80 pp 2-4, 190-191

[Annotation, table of contents and foreword from book by Arkadiy Ventsionovich Barskiy, Mashinostroyeniye, 3225 copies, 192 pages]

[Text] Methods of optimizing parallel computational processes in multiprocess computing systems and computing complexes used in automated control systems are described in the book.

The book is intended for engineers, scientific workers and system programmers studying questions about the design and operation of computer systems.

Contents Page

Foreword

The principles of elimination of overlapping of calculations are a universal method of achieving high productivity and high reliability of computing facilities. The naturalness and universality of overlapping elimination have been the reasons for its use on all levels of the development and use of computer technology--from the planning of equipment overlapping in time the performance of separate operations to the planning of overlapping work of several computers or processors of a multiprocessor computer system on the solution of laborious tasks.

The investigation of problems of overlapping elimination and the development of methods of optimizing it (according to substantiated criteria) permit developing practically realizable principles of computer hardware integration, the organization of controllable parallel computing processes and the logical planning of computer structures and separate equipment.

The introduction of the principles of overlapping elimination stipulated the practical directivity of all material of the present book. The orientation toward the use of computer systems in very complex operating conditions--in ASU--has determined both the circle of tasks under consideration and the selection of flowcharts of computational processes.

Among the available literature on the given question, special note should be made of the book by D. A. Pospelov [4]. The author of the present monograph has tried to avoid to the extent possible dealing with problems which found considerable reflection in that book, for example, problems of segmentation, of the use of languages, storage, the organization of processing conditions, etc. At the same time the author has striven to clarify questions connected with the optimum organization of computational processes, dispatching and the simulation of those processes, the solution of which has been developed and individualized in recent years.

In the book much attention has been given to various methods of organizing the dynamic planning of resources during parallel data processing as the main means of achieving high productivity and reliability of computer systems under variable conditions of realization of the computational process.

In the book a detailed account has been given for the first time of methods of precise solution of the main problems in the elimination of overlapping that arise in the designing of ASU's: optimum marshalling of a computing system to solve a definite set of problems and optimum planning of accomplishment of the given set by the computing system.

The models of parallel computational processes under consideration serve as an adequate illustration of the variety of organizational schemes of those processes and their application, methods of imitating methods of eliminating overlapping and examples of the use of those methods in planning structures of computer systems and compiling parallel programs. The description of various models must contribute to the selection of specific methods of organizing parallel data processing during the development of ASU's.

It is assumed that the construction and use of means of eliminating overlapping in computer systems are possible on the basis of contemporary achievements in the area of the logical possibilities of computer hardware and software: multiprogramming on the basis of the virtual storage, automation of programming, coupling with peripherals, etc. Only the presence of such facilities permits working with adequate freedom with separate tasks, with large program modules and with variants of the commutation of objects in planning the work of a computer system.

[71-2174]

2174
CSO: 1863

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

COMPUTING SYSTEMS

Moscow VYCHISLITEL'NYYE SISTEMY in Russian 1980 signed to press 29 Jan 80 pp 2-4, 302-304

[Annotation, excerpt and table of contents from book by Ivan Vasil'yevich Panfilov and Anatoliy Mikhaylovich Polovko, Izdatel'stvo "Sovetskoye Radio", 8000 copies, 304 pages]

[Text] In this monograph are discussed the fundamentals of the theory of constructing and methods of analyzing and synthesizing, as well as questions relating to the organization of the computing process of multicomputer and multiprocessor computing systems.

This book is intended for scientific personnel and engineers working in the field of studying and designing computers and computing systems. It can also be useful to teachers and students studying computers and computing systems.

Foreword

This monograph is, in the opinion of the authors, one of the first attempts to discuss the key problems in the system design of computing systems and, of course, is not without shortcomings. Comments and advice relating to eliminating them will be received with gratitude.

Chapters 2, 5, 8, 9 and 10 were written by I.V. Panfilov (para 5.5 in conjunction with V.V. Kovalev), chapters 3 and 7 by A.M. Polovko and chapters 1, 4, 6 and 11 by A.M. Polovko and I.V. Panfilov jointly.

The authors express their deep gratitude to Professor and Doctor of Technical Sciences E.V. Yevreinov for his careful reviewing of the manuscript and valuable comments and suggestions which helped improve it.

CONTENTS Page

[77-8831]

8831
CSO: 1863

SET OF GENERAL INDUSTRIAL METHODOLOGICAL HANDBOOK MATERIALS ON THE CREATION OF AUTOMATED CONTROL SYSTEMS (ASU's) AND AUTOMATED DESIGN SYSTEMS (SAPR's)

Moscow KOMPLEKS OBSHCHEOTRASLEVYKH RUKOVODYASHCHIKH METODICHESKIKH MATERIALOV PO SOZDANIYU ASU I SAPR in Russian 1980 signed to press 4 Apr 80 pp 2-3, 7, 118-119

[Annotation, excerpts and table of contents from book published by Izdatel'stvo "Statistika", 15,000 copies, 120 pages]

[Text] Included in the "Set of General Industrial Methodological Handbook Materials (ORMM's) on the Creation of ASU's and SAPR's" are ORMM's on the creation of technological organization ASU's (ASUOT's) and ORMM's on the creation of automated design systems (SAPR's). The materials of this book regulate the structure, content and procedure for carrying out work on the creation of ASUOT's and SAPR's.

For specialists involved in developing, utilizing and improving ASU's and SAPR's in various industries of the national economy.

Approved by V.A. Myasnikov, director, Central Administration for Computer Technology and Control Systems, and member, USSR State Committee for Science and Technology, 20 August 1979.

1.1. The present ORMM's on the creation of technological organization automated control systems (ASUOT's)* supplement the general industrial methodological handbook materials on the creation of automated control systems (ORMM's on the creation of ASUP's [automated enterprise management systems] and ORMM's on the creation of ASUTP's [automated systems for the control of technological processes]) and establish the structure, content and organization of work on creating ASUOT's at industrial enterprises and in production and scientific production associations (enterprises), including work on creating new and developing existing ASUOT's, as well as work on the creation of systems based on existing ASUP's and ASUTP's.

1.2. An ASUOT represents an integrated multilevel hierarchical system combining the functions of an ASUP and ASUTP and enabling the joint functioning, coordinated in terms of purposes, criteria and data processing procedures, of ASUP's and

*Sometimes these systems are called production technology systems.

ASUTP's and other ASU's at an industrial enterprise with the utilization of the required facilities for linking and interaction.

An ASUOT is created at an industrial enterprise for purposes of producing an additional effect greater than the sum of the effects achievable with the independent functioning of individual ASUP's, ASUTP's and other ASU's.

CONTENTS

[77-8831]

8831
CSO: 1863

DESIGN OF DISCRETE AUTOMATION SYSTEMS

Moscow PROYEKTIROVANIYE DISKRETNYKH SISTEM AVTOMATIKI in Russian 1980 signed to press 4 Mar 80 pp 2-4, 232

[Annotation, foreword and table of contents from book by Semen Abramovich Yuditskiy, Agniya Arkad'yevna Tagayevskaya and Tamara Konstantinovna Yefremova, Izdatel'stvo "Mashinostroyeniye", 7000 copies, 232 pages]

[Text] This book is devoted to a discussion of the methodology of designing systems for controlling cyclic processes. The fundamentals of the general theory of cyclic processes are discussed, including models of the controlled system and of the process realized, along with the mathematical apparatus for describing a control automaton (UA); the algorithmic, logic and technical stages of design are discussed, concluding with the formulation of a description of the control algorithm and of the functional and schematic diagrams of the UA. A survey is given of modern equipment for implementing UA's. This book is intended for engineers working on the automation of technological processes and for specialists in the field of applied automata theory.

Foreword

The development and extensive application of various kinds of automatic and automated control systems, including ASU TP's [automated systems for controlling technological processes], have drawn the attention of designers and users of these systems to the theoretical and methodological fundamentals of their structure.

This book is devoted to the theory and practice of the design of one important component of an ASU TP--the discrete control automaton, forming in conjunction with the technological system a system for controlling a cyclic process.

The approach developed over the last two decades to the design of control automata is based on the ideas and methods of the theory of discontinuous (discrete) devices and finite automata discussed in studies by V.M. Glushkov, M.A. Gavrilov, V.G. Lazarev, D.A. Pospelov, A.A. Tal', E.A. Yakubaytis, D. Khafman, R. Miller, etc.

However, as cyclic processes automated within the framework of an ASU TP became more complicated, the classical approach encountered a number of difficulties caused, in particular, by the complexity of a finite automaton description of parallel cyclic processes realized simultaneously in various sections of the technological system and by the lack of regular methods of preparing a design assignment while taking into account the distinctive features of the system, and the like.

In this book an approach to designing control automata is discussed which is oriented toward parallel cyclic processes and which takes into account the properties of the technological system.

This book is based on know-how gained at the Institute of Control Problems with regard to the design of control automata implemented by means of pneumatic equipment. In addition, the authors have strived to demonstrate the feasibility of using the methods which have been developed when employing other equipment, such as electronic integrated microcircuits, programmable logic controllers and electromechanical relays.

In this book discussed in sequence are the fundamentals of the theory of systems for controlling cyclic processes (models of systems and processes and the mathematics for describing the operation of a control automaton) and the general design methodology, illustrated by specific industrial examples and covering the stages of constructing a formalized assignment and designing functional and schematic diagrams for the control automaton. A survey is given of modern equipment for implementing control automata.

Readers wishing to become acquainted with the theoretical and applied aspects of designing control automata should read the book in the order that the material is presented; those who are interested only in design practice in their first acquaintance with the book can omit chapter 2 para 2 and chapter 3 paras 1 and 4.

For the purpose of reading this book it is necessary to have a general acquaintance with the concept of a graph and with the rules of mathematical logic.

Chapters 1, 2, 3 and 5, except chapter 2 para 1 and chapter 5 para 3, were written by S.A. Yuditskiy, chapter 6, except paras 5 and 6, and appendices I and II by A.A. Tagayevskaya and T.K. Yefremova, chapter 2 para 1, chapter 4 para 2 , chapter 5 para 3 and chapter 6 paras 5 and 6 by S.A. Yuditskiy and A.A. Tagayevskaya together, and chapter 4 para 1 by S.A. Yuditskiy and T.K. Yefremova together.

CONTENTS

[78-8831]

8831
CSO: 1863

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300080029-9

UDC 621.391

NEW PUBLICATION ON SPEECH COMPREHENSION SYSTEMS

Moscow PROBLEMY POSTROYENIYA SISTEM PONIMANIYA RECHI in Russian 1980 pp 2, 145

[Annotation, table of contents of book "Problems of Constructing Systems to Understand Speech", edited by Candidates of Technical Sciences G.I. Tsemel' and V.N. Sorokin, Izdatel'stvo "Nauka", 1980, 146 pages]

[Excerpts] This collection of articles is devoted to constructing speech comprehension systems for controlling automata by means of natural language. The levels of speech comprehension systems are reviewed, including analysis of the process of speech formation, analysis of the parameters of speech signals, recognition of words that are merged in pronunciation, and the use of syntactical and semantic information to segment run-on speech and to correct recognition errors.

The book is intended for scientific researchers, engineers, and college students specializing in engineering cybernetics, man-machine communications systems, and processing and converting speech information.

Table of Contents

[63-11176]

11,176
CSO: 1863

UDC 681.32

INPUT-OUTPUT CONTROL IN YeS COMPUTERS

Moscow UPRAVLENIYE VVODOM-VYVODOM V YeS EVM in Russian 1980 signed to press 13 Mar 80 pp 2-4

[Annotation and table of contents from book by Vladimir Iosifovich Gurevich, Izdatel'stvo "Sovetskoye radio", 40,000 copies, 304 pages]

[Text] The book discusses general principles of input-output control in YeS computer operating systems, input-output control devices used in the development of practical programs, questions of data organization on external data carriers and the use of various methods of access to data. The main attention is given to YeS disk operating systems. Information also is presented on YeS operating systems.

The book is intended for specialists engaged in developing YeS computer software, and also to those attending lectures in courses to improve qualifications and VUZ students.

Contents | Page

FOR OFFICIAL USE ONLY

[68-2174]

2174
CSO: 1863

UDC 519.95

NEW BOOK ON PROGRAMMING IN ALGOL

Moscow OSNOVY PROGRAMMIROVANIYA NA ALGOLE in Russian 1980, pp 2, 4

[Annotation and foreword from book "Fundamentals of Programming in ALGOL", by S.A. Abramov and I.N. Antipov, Izdatel'stvo "Nauka", 176 pages]

[Excerpts] This book is devoted to the fundamental elements of programming in ALGOL-60. A systematic description of the language is presented in several stages. There is a discussion of improving problem-solving algorithms and writing them on the basis of rational programs.

The book is intended for a broad range of readers interested in programming in ALGOL-60.

Materials from the authors' book "Programmirovaniye na Uproshchennom Algole" [Programming in Simplified ALGOL] (Moscow, Nauka, 1978) was used in preparing this edition.

Authors' Foreword

ALGOL-60 establishes a certain set of concepts. The removal of several syntactical concepts and the imposition of additional semantic constraints makes it possible to obtain an abbreviated version of ALGOL-60 (sometimes called the simplified version). Conversely, adding new concepts may lead to a broadening of ALGOL-60. Any abbreviated version may be considered as a subset of ALGOL-60. The version named SUBSET-ALGOL may be considered one of the first abbreviated versions of ALGOL-60. This subset was reported, for example, in the journal VYCHISLITEL'NAYA MATEMATIKA I MATIMATICHESKAYA FIZIKA (see No 3, 1975, pp 575-579). The language ALGAMS was published later. It is (with a precision to standard procedures and functions) a subset of the language ALGOL-60 and very similar to the language SUBSET-ALGOL.

In presenting the fundamentals of programming in ALGOL in our book we essentially use a certain simplified version of ALGOL-60. The simplification involves chiefly the removal of certain concepts. We recommend that those who wish to familiarize themselves with ALGOL-60 in its full array refer to S. S. Lavrov's book "Universal'nyy Yazyk Programmirovaniya (ALGOL-60)" [Universal Programming Language (ALGOL-60)] (Third Edition, Moscow, Nauka, 1972). Many exercises for the course in the full ALGOL-60 language can be found, for example, in N. P. Trifonov's book "Sbornik Uprazhneniy po Algolu" [Collection of Exercises in ALGOL] (Moscow, Nauka, 1975).

During preparation of this text we made use of the material in our book "Programmirovaniye na Uproshchennom Algole" [Programming in Simplified ALGOL] (Moscow, Nauka, 1978). The book was written on the basis of our experience teaching programming to students at secondary school No 52 in Moscow. Exchange of opinions with other teachers (associates at the Computing Center of the Academy of Sciences USSR) influenced the content of the book. In addition, we made constant use of consultation and advice from A. A. Abramov, V. M. Kurochkin, N. M. Nagornyy, and V. D. Podderyugin. Important and interesting suggestions and remarks concerning both the selection of the material and the style of presentation were made by corresponding member of the Academy of Sciences USSR S. S. Lavrov.

[63-11176]

11,176
CSO: 1863

UDC 519.6

ALPHA-6 TRANSLATOR IN THE DUBNA SYSTEM

Moscow TRANSLYATOR AL'PHA-6 V SISTEME DUBNA in Russian 1979 signed to press 28 Nov 79 pp 3-8

[Annotation, table of contents and foreword from book by Irina Nikolayevna Anikeyevna, Anatoliy Olegovich Buba, Tat'yana Sergeyevna Vasyuchkova, Svetlana Konstantinovna Kozhukhina, Sergey Eneyevich Kozlovskiy, and Vladimir Ivanovich Shelekhov, A. P. Yershchov, editor, Izdatel'stvo "Nauka", 15,000 copies, 352 pages]

[Text] Contents Page

[71-2174]

2174
CSO: 1863

ABSTRACTS FROM THE JOURNAL 'CONTROL SYSTEMS AND MACHINES'

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 5, 1980, pp 145, 147, 149, 151

UDC 681.3.51./6.42

FUNDAMENTAL CONCEPTS OF RELIABILITY THEORY IN ANALYSIS OF THE OPERATION OF CONTROL PROGRAM PACKAGES

[Abstract of article by Lipayev, V. V., doctor of technical sciences, Moscow]

[Text] Fundamental concepts of probability theory, namely, malfunction, failure and restoration, are studied and formalized when used to analyze the operation of control program packages. The dependence between the reliability characteristics and the time lag of objects controlled by the program package is shown. The concepts of reliable and accurate programs are formalized. Ways of using redundancy to enhance the reliability of programs are described.

UDC 681.3.06.2

PROBLEMS OF COMPUTER RESOURCES ALLOCATION IN SHARED COMPUTER CENTER NETWORKS

[Abstract of article by Lavinskiy, G. V., and Marinchenko, B. V., Ukrainian branch, All-Union State Planning Institute, USSR Central Statistical Administration, Kiev]

[Text] The authors examine problems in the allocation of computer resources. The problem of the location of shared computer centers and terminals is formulated and reduced to a scheme of nonlinear discrete programming with Boolean variables. An algorithm is proposed for the solution of this problem.

UDC 62-52:681.3.06.2

STANDARD MULTILEVEL SYSTEM OF INFORMATION SUPERVISORY CONTROL AND SWITCHING

[Abstract of article by Krivonogov, Yu. A., candidate of technical sciences, and Tsikalenko, V. S., candidate of economic sciences, Ukrainian branch, All-Union State Planning Institute, USSR Central Statistical Administration, Kiev]

[Text] The authors examine the basic concepts of construction and synthesis of standard integrated systems of information-supervisory control and switching at all hierarchic levels of control, as well as the structure of control rooms in such systems.

UDC 681.327.6

OPTIMIZATION OF MAGNETIC TAPE STORAGE PARAMETERS DURING RANDOM DATA FLOW RECORDING

[Abstract of article by Chernyshev, V. M., engineer, Kiev]

[Text] Processes of random data flow recording on magnetic tape are examined. A mathematical model of magnetic tape storage is constructed as a queueing system. Storage operation quality criteria are suggested which can be determined by embedded semi-Markov processes. The optimization problem is formulated.

UDC 65.015.11

DESIGN OF LOGICAL PROCESSORS BASED ON READ ONLY MEMORIES

[Abstract of article by Balashov, Ye. P., doctor of technical sciences, Negoda, V. N., candidate of technical sciences, and Puzankov, D. V., candidate of technical sciences, of the Leningrad Order of Lenin Electrical Engineering Institute imeni V. I. Lenin, Leningrad]

[Text] Questions of the designing of logical processors based on read only memories are examined. A procedure is presented for converting the initial representation of functions into flowcharts by decision tables. Approaches to the minimization of flowsheets and the structural synthesis of logical processors dependent on them are proposed.

UDC 681.3.14/21

USE OF ELEMENT CODE ANALYSIS TO SIMULATE DIGITAL LSI CIRCUITS

[Abstract of article by Isyuk, V. I., and Mikhaylov, B. N., engineers, Novosibirsk]

[Text] A simple method of describing functional, logical and elective MOS circuits to fit problems of logical simulation, one readily realizable with a computer, is proposed. In addition, it does not require development of a translator. A simulation algorithm using the given method of description is examined.

UDC 681.3.06.62

AUTOMATION OF MEMORY ALLOCATION ON GRAPH ALGORITHM MODELS

[Abstract of article by Pogrebnoy, V. K., candidate of technical sciences, Tomsk Polytechnic Institute, Tomsk]

[Text] A method is proposed for solving the problem of automatic memory allocation for algorithms represented as graph models in the language of elementary functions. The method depends on the analysis of the information structure of the graph models, on the application of the results of analysis to construct a matrix of mutual incompatibility in overlaying data components, and on the optimization of their matching in the main computer memory.

UDC 681.3.06:51

COMPUTER-AIDED SYSTEM FOR SIMULATING ALGORITHMS FOR RADIOELECTRONIC ELEMENTS ALLOCATION

[Abstract of article by Gondyul, V. P., instructor, Ein'kovskiy, Yu. F., doctor of technical sciences, and Usatenko, A. N., engineer, of Kiev Polytechnic Institute, Kiev]

[Text] Indicators of effectiveness of algorithms of effectiveness of integrated circuit layout on printed boards are discussed, together with the factors they depend on and their statistical interrelations. A specific structure is suggested for a computer-aided simulation system to investigate layout algorithms.

UDC 681.03.068

DIALOG SYSTEM FOR DEVELOPMENT OF PROGRAMS FOR TABULATED DOCUMENT PRODUCTION

[Abstract of article by Rybakov, S. V., engineer, Sirotkin, V. S. candidate of technical sciences, and Khanykov, V. V., engineer, Moscow]

[Text] Methods are proposed which result in automatic generation of a program yielding an output document at a user's request. Employment of the methods in practice is described. An example of program development is presented.

UDC 681.3.06.069

RECURSIVE SUBPROGRAM REALIZATION IN FORTRAN

[Abstract of article by Livshits, A. B., engineer, Zaporozh'ye branch, All-Union Scientific Research and Design Institute, Scientific Production Association "Soyuztsvetmetavtomatika," Zaporozh'ye]

[Text] A procedure is proposed for representing recursive programs in FORTRAN, one which assures the programming of problems with recursive structure and opens up new possibilities for programming in FORTRAN.

UDC 681.3.068

SELECTION OF A RANDOMIZATION FUNCTION IN ORGANIZING A FILE WITH AN ARBITRARY SCHEDULE

[Abstract of article by Litvinov, V. A., and Ivanenko, V. I., doctors of technical sciences, Institution of Automation, Kiev]

[Text] A generalized parametric randomization function that relies on the general algorith of redundant coding of attributes is offered for a file with an arbitrary randomized structure. Problems of modeling the key-address transformation process and of selecting a specific randomization function are discussed.

UDC 62-52:681.3.06.2

SOME MEANS OF AUTOMATION OF TECHNOLOGICAL ALGORITHM DESIGNING

[Abstract of article by Umbetvayeva, B. A., engineer, Shared Multicomputer Center, USSR State Committee for Publishing Houses, Printing Plants and the Book Trade, Alma-Ata]

[Text] The author examines questions regarding the realization of algorithms for constructing and analyzing the quasi-Polish representation of a graph describing the technology of problem solving in computer-aided management systems.

UDC 681.3.06

REALIZATION OF A SYNTACTICALLY CONTROLLED ANALYZER IN A SYSTEM FOR TEACHING A PROGRAMMING LANGUAGE

[Abstract of article by Drobushevich, G. A., assistant professor, and To Tuan, graduate student, of Belorussian State University, Minsk]

[Text] A procedure of control grammar modification is proposed for effective realization of a syntactically controlled analyzer of teaching systems. A modified control grammar metalanguage is described, as well as the problem of its use in the R-technology of programming.

UDC 681.3.05./94

EDUCATIONAL-PROCESS-ORIENTED SOFTWARE

[Abstract of article by Zvenigorodskiy, G. A., graduate student, Computer Center, Siberian Department, USSR Academy of Sciences, Novosibirsk]

[Text] The requirements are formulated for language means of interaction with computers necessary for the work of students in information science in and out of school. The composition, strategy of design and implementation of software for the educational process in school are defined, as well as the place belonging to that software in the general system software. Various ways of implementing the pupil-oriented programming system now in use and experience in teaching pupils are described.

UDC 681.3.06

MIKALU--A HIGH-LEVEL MICROPROGRAMMING LANGUAGE

[Abstract of article by Valantinas, I. I., candidate of technical sciences, Zhintelis, G. B., candidate of technical sciences, Kanapyatskas, P. N., senior scientific associate, Karbauskas, Yu. Yu., assistant, Karchyauskas, E. K., candidate of technical sciences, and Otas, A. P., senior scientific associate, Kaunas Polytechnic Institute, Kaunas]

[Text] The article presents MIKALU, a high-level microprogramming language developed to describe algorithms for the operation of digital devices without confining itself to specific devices. The language syntax is comprehensively described. Applications of constructions of the language are discussed.

UDC 62-52:681.3.06.2

PALMA DATA BASE CONTROL SYSTEM

[Abstract of article by Glushkov, V. M., academician, Bakayev, A. A., corresponding member, and Kramarenko, R. P., candidate of technical sciences, Special Technological Design Bureau, Production Association, Institute of Cybernetics, UkSSR Academy of Sciences]

[Text] The authors discuss principles underlying the design of the PALMA data base control system, which makes it possible to judge a great variety of views of remote users of fundamental data and to cover all aspects of data base control. A multilevel relational data base model is realized in the system.

UDC 681.3.06/.69

COMPARISON OF VARIOUS DATA STRUCTURES WITH REFERENCE TO RETRIEVAL EFFICIENCY

[Abstract of article by Bartkus, A.-A. T., candidate of technical sciences, and Zhilinskas, R. G., instructor, Vilnius State University]

[Text] Various methods of management of an indexed sequential file are studied. Those methods are compared with one another and their retrieval efficiency is estimated by their activity coefficients and block sizes. Preference fields are established for specific management methods.

UDC 681.3.06:519.2.65

PROGRAM PACKAGE OF A UNIFIED INFORMATION SYSTEM

[Abstract of article by Korostil', Yu. M., engineer, All-Union Institute of Welding Production, Kiev]

[Text] A program package is considered which automatically forms a required information system in accordance with data described in a special manner.

UDC 62-181.4

ADAPTABLE INFORMATION SYSTEMS BASED ON SMALL COMPUTERS

[Abstract of article by Kusrashvili, S. G., engineer, Institute of Cybernetics, UkSSR Academy of Sciences, Kiev]

[Text] An information system based on the YeS-1010 computer is described. The ease with which data are prepared, visual control of stored data, simplicity of editing and feasibility of initiating hierarchic and network search by automatically formulated concept tables enable a large number of users to efficiently use this system in solving problems in various fields of science and engineering.

UDC 681.3.06./94

REAL-TIME PHYSICAL DATA PROCESSING

[Abstract of article by Malinovskiy, B. N., corresponding member, UkrSSR Academy of Sciences, Boyon, V. P., candidate of technical sciences, and Kozlov, L. G., candidate of technical sciences, of the Institute of Cybernetics, UkSSR Academy of Sciences]

[Text] Features of data processing algorithms in real-time systems with instantaneous processes are discussed. Methods enhancing the throughput and speed of data processing facilities in such systems are proposed.

UDC 62-52:681.3.06.2

POSSIBLE TYPES OF INTERACTION IN COMPUTER-ASSISTED RESEARCH SYSTEMS

[Abstract of article by Nakhmanson, M. S., candidate of physical and mathematical sciences, and Shul'meyster, V. M., engineer, of "Burevestnik" Scientific Production Association, Leningrad]

[Text] On examples of specific computer-assisted research systems the authors discuss four types of interaction between the researcher and an instrument on the basis of a mini-computer, interaction which can be accomplished during scientific research: interaction of the "menu selection" type, ordinary dialog, dialog dependent ofn the set of directives and on the BASIC-oriented language. It was found that ordinary dialog and dialog based on the special-purpose set of directives are the most suitable types of interaction in the designing of computer-assisted instruments.

UDC 681.31:(620.16 + 623.544)

SYSTEM FOR AUTOMATION OF RESEARCH INTO MECHANICAL PROPERTIES OF CONSTRUCTION MATERIALS UNDER NEUTRON IRRADIATION

[Abstract of article by Voloshchenko, A. P., Grishko, V. G., Lukashev, V. K., and Aleksandrov, V. Yu., candidates of technical sciences, and Artamonov, V. N. and Dymarskiy. L. M., engineers, Special Technological Design Bureau, Institute of Strength Problems, UkSSR Academy of Sciences, Kiev]

[Text] The architecture of an information control system to automate investigations into the mechanical properties of construction materials under neutron irradiation is discussed. The composition of modules of this system and features of its software structure are defined.

UDC 681.3.51./6.42

FUNCTIONAL ORGANIZATION OF DYNAMIC SIMULATION IN A SYSTEM OF AUTOMATION OF PHYSICAL LABORATORY EXPERIMENTS

[Abstract of article by Vystavkin, A. N., doctor of technical sciences, and Kosachevskaya, L. L., junior scientific associate, of the Institute of Radio Engineering and Electronics, USSR Academy of Sciences (IRE), Novozhilov, N. L., graduate student, Moscow Technical School imeni N. E. Bauman, and Romanovtsev, V. V., candidate of physical and mathematical sciences, IRE, Moscow]

[Text] The principle of functioning of a dynamic simulation subsystem in a measurement-computing complex intended for the automation of several independent experiments is proposed. This principle permits forecasting the state of the system, estimating the degree of its utilization and making a decision on the tactics of investigation in the course of experiments.

UDC 62-52:681.3.06.44

STRUCTURAL SYNTHESIS OF CENTRALIZED CONTROL SYSTEMS FOR NUMERICALLY CONTROLLED MACHINE TOOL SECTIONS

[Abstract of article by Ratmirov, V. A., doctor of technical sciences, ENIMS (Order of the Red Banner of Labor Experimental Scientific Research Institute of Machine Tools), Moscow, Snaksarev, A. M., engineer, Scientific Production Association "Tsentrprogrammsistem," Kalinin, and Yakunin, V. A., candidate of technical sciences, ENIMS]

[Text] Dependences that enable optimizing structures of centralized control systems for work sections equipped with numerically controlled tools are derived by the criterion of total expenses per unit of time and by mathematical apparatus of queueing theory. Fields of efficient use of differently structured control systems of numerically controlled machine tool sections are defined.

UDC 62-52:681.3.06.44

OPERATING DATA EXCHANGE SYSTEM IN COMPUTER-AIDED PRODUCTION PROCESS CONTROL SYSTEM EMPLOYED AT A HYDROELECTRIC POWER STATION

[Abstract of article by Shestunov, E. A., engineer, Gidroproyekt Scientific Research Station, Vyshgorod]

[Text] Methods and results of structurization as well as languages and algorithms of functioning of the operating control system of a computer-aided production process control system employed at a hydroelectric power station are described. The formulated principles of organization of those subsystems discussed in the article (data representation, the operator's panel, documentation, operations check and reference-instruction information) of control and representation languages realized by means of a vocabulary display, the graphic control panel and a standard display are applicable to the control of other production processes and can be used as standard solutions.

UDC 681.3.06:622.26919

DESCRIPTION OF A DYNAMIC IDENTIFICATION PROGRAM PACKAGE

[Abstract of article by Meleshko, V. I., candidate of technical sciences, NAZYROVA, V. P., engineer, and ZABARA, I. M., junior scientific associate, Khar'kov State University, Khar'kov]

[Text] A description is given of the structure, information and functional subsystems, main modules and a problem-oriented language of a dynamic identification program language.

UDC 681.3.51./6.42

PACKAGE OF APPLIED PROGRAMS FOR MODELING THE MOVEMENT OF PLANE GEOMETRIC OBJECTS IN A COMPUTER-AIDED DESIGN SYSTEM

[Abstract of article by Nikitin, A. I., doctor of technical sciences, Institute of Cybernetics, UkSSR Academy of Sciences, Kiev, and Popov, V. L., engineer, and Pustyl'nik, G. M., junior scientific associate, All-Union Scientific Research Institute of Electrical Machine and Apparatus Building, Khar'kov]

[Text] Algorithms and principles of realization of a package of applied programs for modeling the movement of plane geometric objects are described. The package is used in a computer-aided design system when arranging construction elements with respect to given constraints.

UDC 681.327.634

ORGANIZATION OF INPUT-OUTPUT AND EXTERNAL FLEXIBLE MAGNETIC DISKS IN SM COMPUTERS

[Abstract of article by Borisov, A. V., Sosnovskiy, A. K., Ozhiganov, Yu. M., and Cherepanov, V. A., engineers, "Elektronmash" Production Association, Kiev]

[Text] Problems in the organization of the external flexible magnetic disk storage for SM-3 and SM-4 control computer complexes are examined. Instructions and the structure of the main registers of the controller of such disks are described.

[54-2174]

2174
CSO: 1863

END