JPRS 83420

6 May 1983

USSR Report

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

No. 69

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.

PROCUREMENT OF PUBLICATIONS

JPRS publications may be ordered from the National Technical Information Service (NTIS), Springfield, Virginia 22161. In ordering, it is recommended that the JPRS number, title, date and author, if applicable, of publication be cited.

Current JPRS publications are announced in Government Reports Announcements issued semimonthly by the NTIS, and are listed in the Monthly Catalog of U.S. Government Publications issued by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

Correspondence pertaining to matters other than procurement may be addressed to Joint Publications Research Service, 1000 North Glebe Road, Arlington, Virginia 22201.

JPRS REPORTS

Japan Report
Korean Affairs Report
Southeast Asia Report
Mongolia Report
Near East/South Asia Report
Sub-Saharan Africa Report
West Europe Report
West Europe Report: Science and Technology
Latin America Report

USSR

Political and Sociological Affairs
Problems of the Far East
Science and Technology Policy
Sociological Studies
Translations from KOMMUNIST
USA: Economics, Politics, Ideology
World Economy and International Relations
Agriculture
Construction and Related Industries
Consumer Goods and Domestic Trade
Economic Affairs
Energy
Human Resources
International Economic Relations
Transportation
Physics and Mathmetics
Space
Space Biology and Aerospace Medicine
Military Affairs
Chemistry
Cybernetics, Computers and Automation Technology
Earth Sciences
Electronics and Electrical Engineering
Engineering and Equipment
Machine Tools and Metal-Working Equipment
Life Sciences: Biomedical and Behavioral Sciences
Life Sciences: Effects of Nonionizing Electromagnetic Radiation
Materials Science and Metallurgy

EASTERN EUROPE

Political, Sociological and Military Affairs
Scientific Affairs
Economic and Industrial Affairs

CHINA

Political, Sociological and Military Affairs
Economic Affairs
Science and Technology
RED FLAG
Agriculture
Plant and Installation Data

WORLDWIDE

Telecommunications Policy, Research and Development
Nuclear Development and Proliferation
Environmental Quality
Epidemiology

FBIS DAILY REPORT

China
Soviet Union
South Asia
Asia and Pacific
Eastern Europe
Western Europe
Latin America
Middle East and Africa

To order, see inside front cover

JPRS 83420

6 May 1983

USSR REPORT

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

No. 69

CONTENTS

CONFERENCES

PUBLICATIONS

UDC 681.3

NEW HIGH-PERFORMANCE ELEKTRONIKA-79 MINICOMPUTER AND EFFICIENCY OF USE IN SCHEMATIC DESIGN SYSTEMS OF LARGE INTEGRATED CIRCUITS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 14 Jan 82, after revision 15 Jul 82) pp 105-108

[Article by Candidate of Technical Sciences Igor' Leonidovich Talov, Kiev, engineer Vladimir Vasil'yevich Plotnikov, Voronezh, Candidate of Technical Sciences Vyacheslav Yegorovich Mezhov, Voronezh, and senior scientific associate Gennadiy Grigor'yevich Bondarovich, Kiev Polytechnical Institute]

[Text] The new high-performance Elektronika-79 minicomputer (see figure) designed to solve a wide range of problems in automation of control, design and efficiency processing of large data files, has now been developed.

The new minicomputer works with numbers represented in formats with fixed and floating point. The computer instruction system includes instruction in three formats (nonaddress for control and one- and two-address for calculations) and permits eight methods of addressing. The set of instructions consists of 137 basic instructions. A total of 88 instructions is used to process numbers with fixed point with word length of 16 and 32 bits, while 46 instructions are used for numbers with floating point with word length of 32 and 64 bits.

A separate processor connected to the central processor by an internal bus system performs operations with numbers with floating point. The speed of the computer reaches three million operations per second and the main memory capacity is 4 Mbyte. The main memory of the computer is based on dynamic type BIS [large integrated circuit] that require periodic regeneration of the contents, which increases the access time to it and reduces the speed of the computer. A high-speed associative memory with access cycle of 300 ns, the so-called CACHE memory, with small capacity (2 Kbyte), has been introduced to increase the speed of the computer between the main memory and the central processor.

The central processor and interfaces of the external memory and peripheral data input-output devices are connected by a common bus system--by a single channel fully compatible with the channel of the Elektronika 100-25 minicomputer. However, data are exchanged between the processor and external memory devices (three groups of external memory devices can be connected to the computer) through a separate 32-digit channel, which guarantees high speed of

exchange, while only the control registers are accessible from the direction of the unified channel. Access to the main memory from the processor is also gained through a separate 22-digit memory channel.

The central processor communicates with the main memory channel and the unified channel by using a memory dispatcher which converts the 16-digit program address to an 18-digit unified channel address and 22-digit main memory address. Access is gained to the main memory from the peripheral devices through a channel address converter.

Thus, three address zones: a 22-digit with access to the main memory, an 18-digit with access to the channel and 16-digit virtual memory that use programs, are used in the minicomputer. The three operating modes of the central processor (internal, supervisor and user), 16 general-purpose registers, large main memory capacity and also the capability of connecting a large number of external memory devices permit organization of multiprogram operation and achievement of high computer productivity in multistation systems.

The computer includes:

a processor based on bipolar circuits with Schottky diodes and designed for storage, processing and control of data flows;

a main memory based on integrated circuits of the K565 or K581 series, which are dynamic type storage devices; the minimum memory capacity is 256 Kbyte;

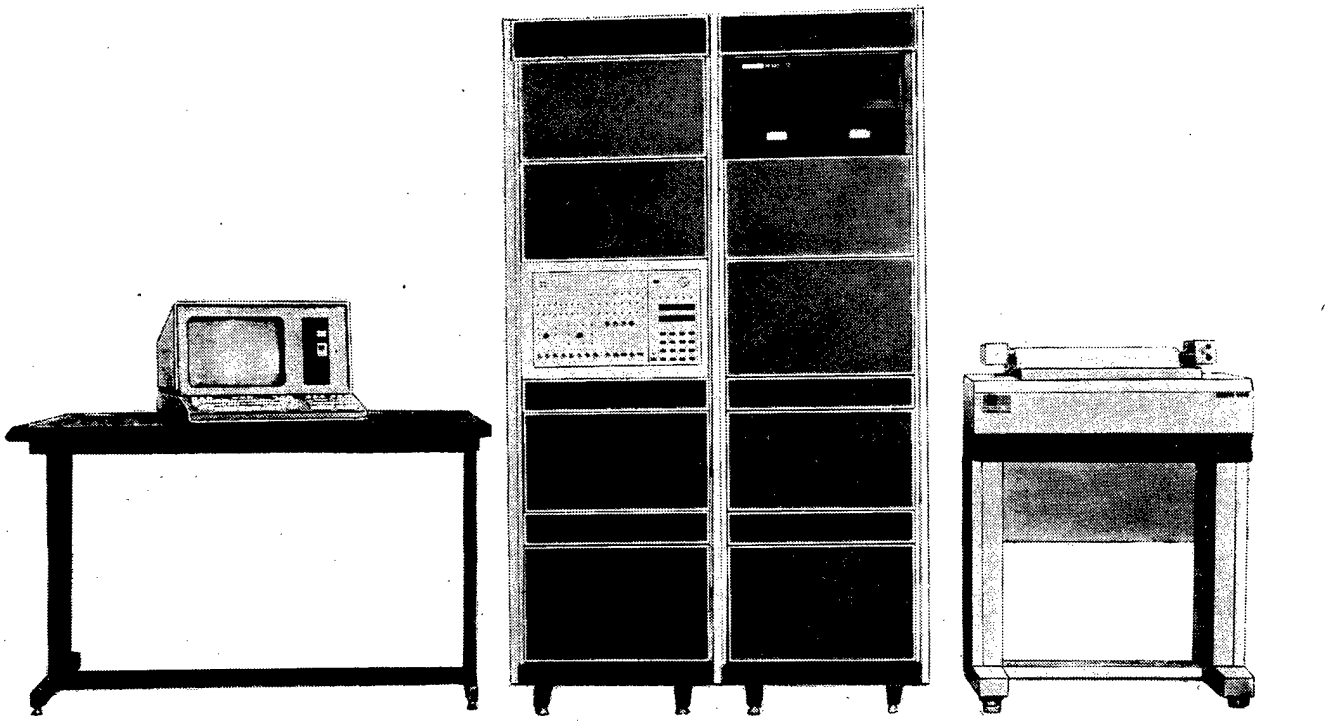
external memory devices: SM-5400 disk unit (subsequently the YeS-5061) and the IZOT-5003 tape unit;

data input-output peripheral devices that include a floppy disk unit of the PLx-45D type or the GMD-70 Elektronika type, a 15IE-00-002 or 15IE-00-013 alphanumeric display, an SM-6204 papertape station and a DZM-180 or DARO-1156 printer.

The Elektronika-79 is program-compatible with the smallest models of the Elektronika 100-25 and the Elektronika-60M and can also be used efficiently with the SM-3 and SM-4 computers in multiprocessor and multistation design, control and scientific research automation systems.

The results of comparing the efficiency of using the Elektronika-79 minicomputer and the 15 UT-4-017 circuit design complex for solving problems of circuit design of electronic equipment parts are presented in the given article. The hardware and software systems for circuit design are considered in detail in [1-3].

Three methods of input data preparation, entry and processing on circuits of the functional units of BIS were used for an experimental check of the efficiency of using the 15 UT-4-017 system in solving routine circuit design problems.



The first method was comparison of drawings of the functional units of BIS, preparation of input data based on the rules of YaZOS-4 [4] or SAMRIS [5] input languages, punching the data on punchcards or papertape, entry of it into the BESM-6 computer and also program and visual check of the entered data, printout of the input data and checking and editing of it.

In the second method, complete information on the functional diagrams of the BIS units was prepared in text form according to YaZOS-U language and was entered from the alphanumeric display at the circuit designer's work position. Input, checking and editing of information were carried out by using developed software in the interactive mode [3].

The third method (like the second method) included preparation of textual information on the circuits and entry of it from the alphanumeric display according to the rules of YaZOS-U input language. Graphic data was entered from the prepared drawings of the functional diagrams of BIS using the EM-719B graphic data coder. All the software for entry, checking and editing of textual information by using the alphanumeric display was used in the given method, while the 15IG-001 graphic memory display and EM-729A device for controlling the position of the marker of this display [3] was used for graphic data.

Comparison of the first method, widely employed at present, and of the developed second and third methods of data preparation, entry and processing on the functional circuits of BIS made it possible to achieve an objective evaluation of the efficiency of developed software-hardware of the circuit designer's work position.

The efficiency of the developed methods and of the hardware-software was determined experimentally on six functional circuits of BIS of various types of technologies, selected for this purpose, with complexity from 5 to 394 components and from 7 to 139 assemblies.

Table 1.

Complexity of Circuits (Components/Assemblies)	Data File Preparation Time		
	First Method	Second Method	Third Method
5/7	25 min	9 min	11 min
8/7	2 hr 42 min	50 min	28 min
76/52	5 hr 45 min	2 hr 25 min	2 hr 4 min
89/55	10 hr 20 min	3 hr 52 min	2 hr 17 min
165/77	8 hr 50 min	5 hr 52 min	4 hr 38 min
394/139	12 hr 10 min	8 hr 54 min	7 hr 38 min

The time expenditures to perform operations by the three methods of textual and graphic data preparation, input and processing by the operator are presented in Table 1. Comparison of the results shows that the second and third methods permit the laboriousness of performing the operations to be reduced two-fifths to one-third. A considerable reduction of laboriousness, especially with the third method, is achieved with an increase of the complexity of the BIS functional diagrams. This is explained primarily by the fact that the operations are performed in the interactive mode. When entering textual information on the circuits, automatic prompting of the operator's actions and program and visual checking of entered data are carried out. The user is freed of the need to code the drawings and of manual preparation of more complex files on circuit switching. This information is formulated automatically.

The operational editing programs guarantees simplicity and high rate of error correction in the input data. The fact that the errors in the data are determined at earlier stages of the BIS design process is of considerable importance. Automatic formulation of the higher capacity and more complex switching files (from the viewpoint of verification) guarantees a high percentage of data reliability and reduces the probability of losses of machine time on calculation of the electric characteristics of the functional circuits of BIS having undetermined errors in the input data.

The electric characteristics of the functional units of BIS were analyzed using a complex of PRANIS programs (written in Fortran-4 language), the main characteristics of which are presented in [3]. Programs for analysis of the static and dynamic characteristics of the functional units of BIS for the

BESM-6 were used as the bases of this complex [6, 7], which permitted correct comparison of the results during the experiment on the Elektronika 100-25 and Eleketronika-79 computers to similar results of calculation on the BESM-6 computer.

The following sequence of conducting the experiment was used: the functional circuits of BIS of different classes were selected, the input information was entered into the computer and processed, the static and dynamic characteristics were calculated on the indicated computers, comparative analysis of the calculating expenditures to calculate the static and dynamic characteristics of the selected functional units of the BIS was carried out and recommendations were worked out on the use of the circuit designer's work position in the autonomous mode and for organization of a circuit design system based on the new minicomputer.

The following circuits were selected to conduct the experiment:

an AND-NOT circuit based on bipolar transistors having five components and seven assemblies;

an AND-NOT circuit based on MDP devices containing eight components and seven assemblies;

an equivalence circuit of the functional unit of the BIS with injection power supply having 76 components and 52 assemblies;

an equivalence circuit of the functional unit of the BIS for control of the microcomputers on the injection devices, containing 89 components and 55 assemblies;

an equivalence circuit of the microcomputer unit based on injection devices, containing 165 components and 77 assemblies.

The time expenditures for calculation of the static and dynamic characteristics of the enumerated circuits on the BESM-6 (first lines), Elektronika 100-25 (second lines) and Elektronika-79 (third lines) computers, respectively, are presented in Table 2.

Summarizing the results of experimental evaluation of the efficiency of using the 15 UT-4-017 system and of the high-speed Elektronika-79 minicomputer to solve routine and engineering problems of circuit design of BIS, one can conclude the following:

the new high-speed Elektronika-79 minicomputer permits solution of complex scientific and technical problems of automation of electronic equipment parts design;

the developed methods of processing textual and graphic information on circuits using the 15 UT-4-017 complex guarantee a two-fifths to one-seventh reduction of BIS design deadlines compared to the methods used.

Table 2.

Complexity of Circuits (Components/ Assemblies	Number of Oscillators	Number of Oscillator Drops	Simulation Time, ns	Number of Points of Transfer Characteristics	Time of Calculation of Characteristics	
					static	dynamic
5/7	2	3	2,000	10	18 s	36 s
					1 min 28 s	1 min 57 s
					53 s	1 min 13 s
8/7	3	7	400	11	13 s	35 s
					1 min 3 s	1 min 52 s
					38 s	1 min 8 s
76/52	12	16	1,600	6	21 min 59 s	50 min 53 s
					1 hr 23 min 18 s	5 hr 33 min 25 s
					28 min 57 s	2 hr 00 min 3 s
89/55	8	13	1,100	21	18 min 20 s	42 min 17 s
					1 hr 2 min	2 hr 54 min 17 s
					40 min	2 hr 10 min 11 s
165/77	5	6	1,000	13	15 min 36 s	36 min 28 s
					1 hr 2 min	3 hr 17 min 25 s
					34 min 55 s	1 hr 17 min 29 s

BIBLIOGRAPHY

1. Mezhov, V. Ye., N. L. Ratmirov, I. L. Talov and B. L. Tolstykh, "Using the Elektronika 100-25 Minicomputer in an Automated Circuit Design System," ELEKTRONNAYA PROMYSHLENNOST', No 10, 1978.

2. Mezhov, V. Ye., N. L. Ratmirov, I. L. Talov and B. L. Tolstykh, "The Software of the 15 UT-4-017 System," ELEKTRONNAYA PROMYSHLENNOST', No 6, 1979.

3. Mezhov, V. Ye., I. L. Talov and Yu. N. Chernyayev, "The Applied Programs of the 15-UT-4-017 Circuit Design System," ELEKTRONNAYA PROMYSHLENNOST', No 7, 1980.

4. Tolstykh, B. L. and V. Ye. Mezhov, "The Graphic-Symbolic Input Language of the 15 UT-4-017 Interactive Graphic System," ELEKTRONNAYA TEKHNIKA, MIKROELEKTRONIKA, No 4, 1979.

5. Vasenkov, A. A., G. G. Kazennov, Yu. N. Belyakov and P. P. Avrashkov, "The SAMRIS Automated Simulation and Integrated Circuit Calculation System," MIKROELEKTRONIKA, No 6, 1976.

6. Nikishin, V. I., I. L. Talov, V. V. Yur'yev and V. M. Mezhov, "A Universal Program for Analysis of the Static Characteristics of EPAS-U Nonlinear Electronic Circuits," TRUDY INEUM, No 58, 1976.

7. Nikishin, V. I., I. L. Talov, V. Ye. Mezhov et al, "An Efficient Program for Analysis of the Dynamic Characteristics of Electronic Circuits and Use of It to Analyze Bipolar Integrated Circuits with Injection Power Supply," Loc. cit.

6521
CSO: 1863/71

UDC 681.327.11:621.3.049.771.14

PROBLEM OF INTERNAL INTERFACE IN MICROPROCESSOR SYSTEMS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6 , Nov-Dec 82
(manuscript received 3 Mar 81, after revision 16 Jun 82) pp 17-24

[Article by Doctor of Technical Sciences Mark Petrovich Gal'perin, engineer Valeriy Viktorovich Gorodetskiy and engineering Anatoliy Fedorovich Dryapak, LOEP Svetlana, Leningrad, and graduate student Vladimir Nikolayevich Oginskiy, Leningrad Institute of Precision Optics and Mechanics]

[Text] Introduction. The problem of working out a unified set of rules for integration of independent computer systems (processor, memory devices and input-output devices), connection of peripheral devices and finally combining several computers into a unified system is an important one for any class of modern computers.

With the appearance of the microcomputer, the problem of the interface became even more timely, having touched on a considerably wider range of sectors of science and technology. First, the microcomputer is the largest class of modern computers. Second, microcomputers are used mainly as built-in devices, with regard to which solution of the interface problem largely determines the process of design of an extremely wide range of devices, machine tools, assemblies and systems.

The importance of the interface problem to microcomputers is also determined by the fact that this problem is divided into two parts: interfaces for microcomputers (in the traditional postulation for computer technology) and "microcomputers for interfaces," i.e., the use of microcomputers as a general-purpose programmable components for realization of protocols for integration of various types of equipment (of several measuring devices into a unified system, of several automated machine tools into a unified section controlled from a central computer, data transmission apparatus that forms some communications network together with communications channels and so on).

Consideration of this complex and dual problem and also solution of the special problem of selecting the interface when developing a microprocessor set of BIS [large integrated circuit] with increased functional complexity are unthinkable without clear definition of the concept of interface. We shall

understand an interface as some set of rules (protocols) that determine integration of electronic devices of different level of complexity.*

The entire set of interfaces of microcomputers can be divided into external and internal.

The external interfaces of microcomputers, as follows from the name itself, determine integration of the microcomputer with an external medium--with peripheral devices, sensors and actuating mechanisms, control and display members and the data transmission channels of devices, assemblies and systems in which (or jointly with which) the microcomputer (user interfaces) operates.

Internal interfaces of microcomputers determine the integration of the constituent parts of the microcomputer with each other into a system of specific configuration. Two groups of internal interfaces can be distinguished at the given phase of development of microprocessor (MP) and microcomputer technology:

BIS interfaces of the microprocessor set (intracard interfaces);

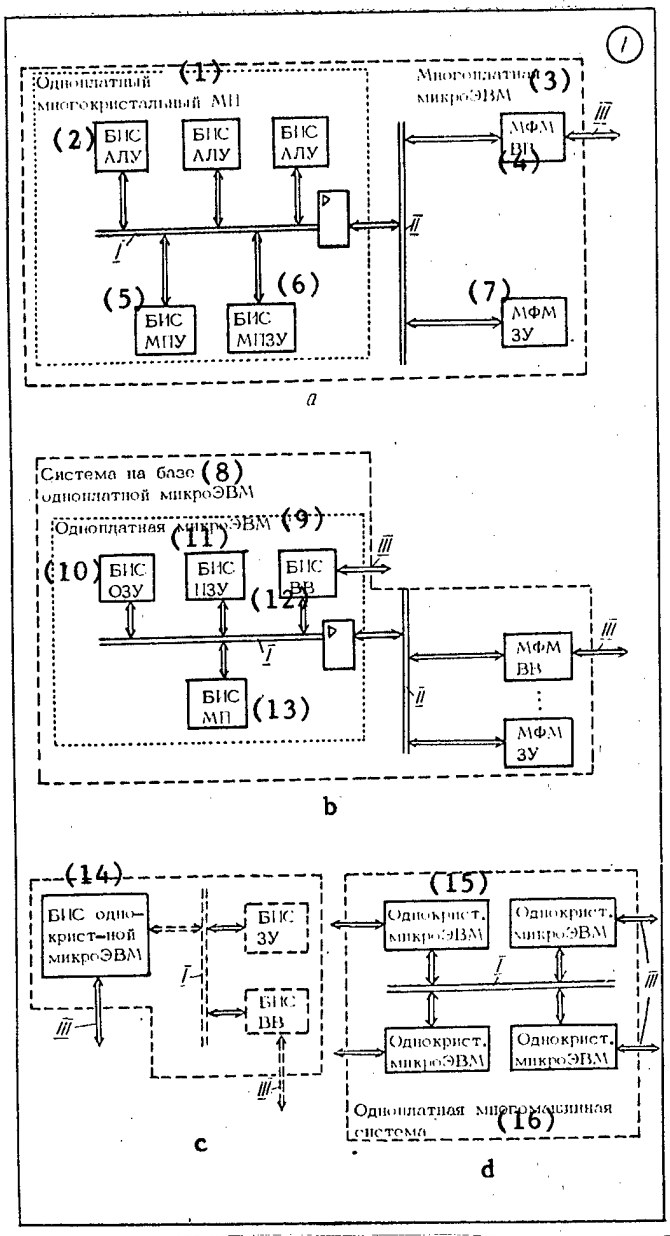
interfaces of microprocessor functional modules (intercard interfaces).

The various types of interfaces are presented in Figure 1.

Multichip microprocessors are usually constructed on the BIS of microprocessor sets produced within a single family and, therefore, no additional equipment is required to realize a "intraprocessor" interface. Due to the sufficient number of leads with low digit capacity of the ALU [arthimetic-logic unit] sections, the intraboard interface of these microprocessors can be more powerful than that of single-chip microprocessors and may require fewer supplementary integrated circuits for integration with ZU [memory] BIS and UVV [input-output device]. However, if a multichip microprocessor, memory BIS and UVV cannot be arranged on one board, they are integrated at the level of the interboard interface (Figure 1, a).

When designing single-board microcomputers based on single-chip microprocessors, there must be a different number of supplementary equipment to integrate the microprocessors with memory BIS and UVV (Figure 1, b). Thus, approximately 20 SIS [very high-speed integrated circuit] housings are required to integrate the Intel-8008 microprocessor to external memories and UVV, only six supplementary TTL [transistor-transistor logic] IS, part of which performs only the role of buffer amplifiers, are required for integration of the Intel-8080 microprocessor, in which the greater part of the supplementary equipment required for the Intel-8008 microprocessor is realized in the composition of the microprocessor BIS itself. At the same time, supplementary circuits are not required at all for integration of the 6800 microprocessor (of the

*The aggregate of firmware understood as an interface by most authors [1] is one of the possible physical realizations of the interface as a protocol for integration of devices.



Examples of Interfaces: a--in multiboard microcomputer; b--in system based on single-board microcomputer; c--in single-board system based on single-chip microcomputer; d--in single-board multimachine system based on single-chip microcomputer (I--intraboard interface; II--external interface of Unibus type; III--external input-output interface)

Key:

1. Single-board multichip microprocessor
2. Large integrated circuit of arithmetic-logic unit

[Key continued on following page]

[Key continued from preceding page]:

3. Multiboard microcomputer
4. Input-output microprocessor functional module
5. BIS of microprocessor device
6. BIS of microprocessor memory
7. Microprocessor functional module of memory
8. System based on single-board microcomputer
9. Single-board microcomputer
10. BIS of main memory
11. BIS of ROM
12. Input-output BIS
13. Microprocessor BIS
14. BIS of single-chip microcomputer
15. Single-chip microcomputer
16. Single-board multimachine system

Motorola Company) with the memory BIS and UVV [2]. It is obvious that the volume of supplementary equipment required to design one or another computer configuration based on BIS of various microprocessor sets is the most important user characteristic of these sets, which in turn is wholly determined by the intraboard interfaces used for the BIS of this set. Moreover, just as important an indicator is the amount of supplementary equipment for conversion from an intraboard to interboard interface.

Design of single-chip microcomputers with the capability of expanding the internal resources (volume of the main memory and of the input-output devices) requires that the greater part of the leads be used for the intraboard interface (Figure 1, c). In this case a single-chip microprocessor actually degenerates into a microprocessor. An example is the single-chip K1801VYe1 microcomputer, which has a systems mainline controller for integration with the memory BIS and UVV and an external interface in the form of a single sequential channel [3].

Information can be exchanged between the microcomputer through sequential channels (so-called weak communications) and the greater part of the leads to realize the external interface when developing single-chip multimachine systems based on single-chip microcomputers (Figure 1, d).

Internal interfaces of the microcomputer. These interfaces are not dependent on the range of application of the microcomputer and are selected with regard to the following, sometimes contradictory requirements:

achievement of maximum speed during data exchange between the constituent parts of the microcomputer;

low response time to interrupt;

elimination or minimization of the number of circuits of supplementary equipment required to integrate the constituent parts into a unified system;

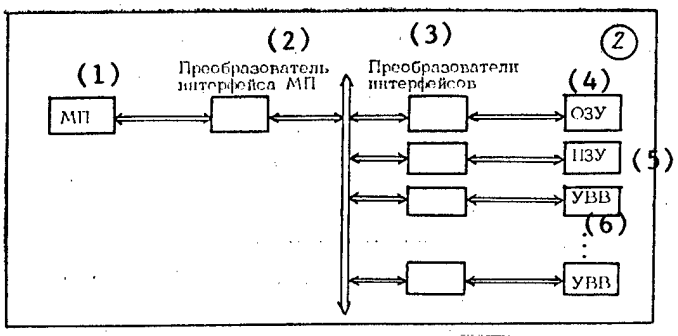
minimization of the number of BIS leads used for the intraboard interface and of the number of plug contacts used for the interboard interface;

the presence of functional capabilities that permit direct access to the memory without supplementary equipment, arbitrage of the systems bus, interrupt priorities and so on;

electric and design compatibility of the interfaces;

independence of the interface from development of the production base.

Data exchange occurs between the microprocessor, memory and UVV in any microcomputer. Regardless of the complexity, three types of signals always circulate in the system--data, address and control signals. These signals are either transmitted each through its own bus system or are spatially combined and separated in time. When the BIS (microprocessor functional modules) with different intraboard or interboard interfaces are integrated with each other, supplementary equipment that performs the role of interface converters is required (Figure 2).



Design of System Based on BIS (Microprocessor Functional Modules) with Different Intraboard and Interboard Interfaces

Key:

1. Microprocessor
2. Interface converter of microprocessor
3. Interface converters
4. Main memory
5. ROM
6. Input-output device

It is obvious that design of microcomputers does not require supplementary equipment when realizing interface converters as the BIS themselves (microprocessor functional modules), but this means that the task of the microcomputer developers is simplified to the maximum extent. Hence, the need to standardize the internal interfaces immediately follows, which guarantees:

the capability of integrating the BIS of microprocessor sets into a unified system without complex analysis of the time and electric characteristics of each BIS and without supplementary equipment;

the compatibility of newly developed BIS (microprocessor functional modules) with existing and future microprocessor products.

the compatibility of microprocessor BIS of different manufacturers.

Moreover, creation of standards for the internal interfaces of microcomputers will guarantee widespread introduction and study of them, which will make it possible to make the use of microcomputers, microprocessor functional modules and the BIS of microprocessor sets just as simple and widely used as small- and medium-scale integrated circuits are now used.

How then should the internal interfaces be assigned for different devices (microprocessors, main memories and input-output devices) so that these devices are integrated with each other without supplementary equipment? We feel that the more feasible is being given the composition of the systems bus and the sequence of the time and electric characteristics of signals on it, which determines the functional and electric characteristics of the internal interfaces of devices directly connected to the bus.

Different companies have developed a number of standards for systems buses that determine internal interfaces of different levels (intraboard and interboard). However, the characteristics of most of these buses were established historically: microprocessor sets of basic BIS (sets of microprocessor functional modules) were initially developed, it then became necessary to develop BIS or such modules for expansion of the functional capabilities of systems and as a result of this, the need arose for simple integration of newly developed BIS (modules) with existing systems. The composition of existing buses was established with some expansion for future applications, the time and electric characteristics of the signals on the bus were determined and interfaces of newly developed BIS (microprocessor functional modules) were accordingly designed).

Advance standardization of the internal interfaces of microcomputers is feasible with regard to the interconnection of intra- and interboard interfaces and the established quite specific tendency to convert from multiboard to single-board microprocessor systems and from single-board configurations to realization of them on a single chip, in which the intraboard interface becomes closer to the interboard interface in its functional capabilities.

The large nomenclature of manufactured microprocessors and microcomputers and also the large number of developer companies determined the appearance of the most diverse company and intercompany standards for buses, the most widespread among which are: Microbus [4], Multibus [5], S-100 [6], Q-bus (LSI-11) [7], Versabus [8], STD-bus (Pro-Log), Z-bus (Zilog), TM-990 bus (Texas Instruments) [9, 10].

Let us consider the main limitations of these buses, taken into account in development of the intraboard interface for a microprocessor set of BIS of increased functional complexity.

Bus configuration. Buses can have combined (Q-bus and Z-bus) and separate address and data lines (S-100, Multibus, Microbus and Versabus). Separate buses permit address and data transmission to be combined in time, but reuqire a greater number of leads for the internal interface. If this is insignificant

due to the rather large number of plug contacts for the interboard interface, then the use of separate address and data buses limits the digit capacity of these buses or of the control bus for the intraboard interface of the BIS whose housing usually has 40-48 leads [11].

An increase of the digit capacity of microprocessors and microcomputers requires expansion of both the data and address buses. Most buses permit exchange of both 8- and 16-digit data between the devices of the system by introduction of supplementary control signals. The Versabus, for example, provides operation with 8-, 16- and 32-digit data, which permits devices with different digit capacity to be used in the same system. There is also the capability of expanding the address bus (up to 18 in the Q-bus, up to 24 in the S-100 and up to 32 digits in the Versabus) with the lines reserved for this purpose.

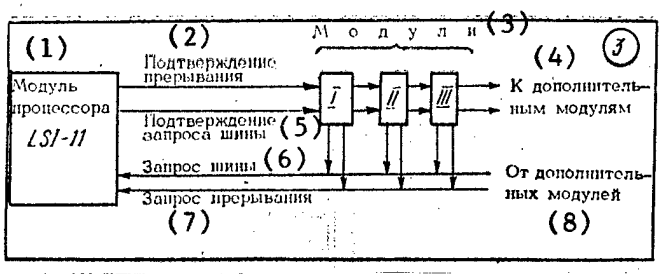
Bus arbitrage. Data exchange is organized on most buses on the "master-slave" principle, in which the master device takes control of the bus on itself. The presence of several devices having the capability of becoming master devices in the system gives rise to the problem of distributing the bus resources among these devices (bus arbitrage). This arbitrage is accomplished differently in each bus.

The use of a direct-access controller to the memory or design of a supplementary circuit is assumed by the structure of the microbus in these cases. The number of direct-access interrogation lines to the memory and confirmation of it is not specially stipulated.

A four-digit bus that permits one to determine which of the 16 possible master devices in the system can take over control of the bus at a given moment is used in the arbitrage system of the S-100 bus. The device that has established the highest-order code on the arbitrage bus can take possession of the bus if there is a signal transmitted by the central processor authorizing takeover of the bus in response to a demand for takeover. This method is rather convenient, permits organization of a change of priorities of devices during completion of programs, but limits the total number of devices in the system and requires a rather complex hardware part and requires a greater number of leads from the BIS when used for the intraboard interface.

Arbitrage in the Q-bus is simpler (Figure 3), where the device located closest to the "daisy chain" with respect to the processor has the highest priority. The device with the highest priority passes the request confirmation signal of the bus to the device with the lowest priority only if it itself does not require control of the bus. Arbitrage is also accomplished in a similar manner in systems based on the Multibus. This method, although it does not have flexibility in designation of priorities, is simple to realize and requires a small number of lines and when used for the intraboard interface, may fully satisfy the design of complex single-board systems with unlimited number of master devices.

There are five levels of bus inquiry in the Versabus from BR4 (highest priority) to BR0 (lowest priority). Organization of the priority scheme is the



Organization of Priority Circuits in System Based on Q-bus

Key:

1. Processor module
2. Confirmation of interrupt
3. Modules
4. To supplementary modules
5. Confirmation of bus inquiry
6. Bus inquiry
7. Inquiry
8. From supplementary modules

same on each level as in the Q-bus. Having received an inquiry, the arbiter compares its priority to that of the inquiry of the current master device and if the received inquiry has higher priority, it transmits a bus cleaning signal. After the last exchange cycle in the bus has been completed, the current master device frees the bus. The arbiter transmits the signal to make the bus available on one of five BR0-BR4 lines according to the level of priority of the incoming request. The device that has received the BQ signal becomes the master. This version of taking over the bus is unacceptable for the intraboard interface because of the need for a large number of leads from the microprocessor BIS.

Organization of interrupts. The majoriy of microprocessor systems is designed to operate in real time [12]. Therefore, the presence of an interrupt mechanism and also determination of priorities during operation of several interrupt sources in the system are important characteristics of the bus.

There are two main methods of interrupt: interrupt with polling and interrupt by vector [2]. In the first case, when the interrupt signal arrives, the central processor completes the instructions and transmits the signal of confirmation of readiness for interrupt processing, after which all the authorized interrupt sources are polled. Conversion to the corresponding interrupt service program is made after detection of the interrupt source. Polling can be carried out either through the software or hardware method. The priority of the interrupt source is determined by its location in the polling sequence. In the case of vector interrupt, the processor is freed of determination of the interrupt source address and immediately receives instructions for the corresponding service program.

The interrupt signals can be prohibited by program in both cases by using masks (can be masked). This permits control of such interrupt service characteristics as the waiting time and the time the demands are in the system [13] and also dynamically changes the level of the interrupt request priority.

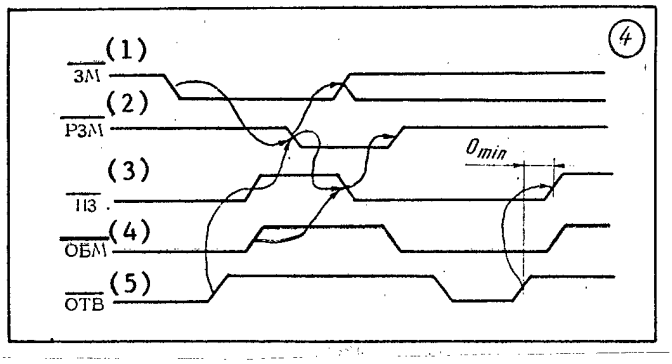
Although the vector interrupt method requires supplementary hardware, it is used in most of the mentioned buses since it is higher speed and requires a small number of control signals. It can be used for the intraboard interface.

The INTERBIS interface. The Soviet electronics industry is developing three main families of 16-digit microcomputers: the Elektronika-S5, the Elektronika-60 and the Elektronika-NTs. Investigations are being conducted to standardize the interfaces and software of these devices.

The characteristics of the interboard interface of the Elektronika-60 family are used as the unified functional characteristics. However, BIS having different intraboard interfaces are produced within different families, which makes it difficult to develop systems based on BIS of different families, integration of intraboard and interboard interfaces in these systems and also results in nonproduction expenditures for development of BIS of the same type.

The INTERBIS standard for the intraboard interface of the BIS of microprocessor sets has been suggested to eliminate the indicated deficiencies, within which the composition of the systems mainline, the order and types of interactions of the BIS connected to the mainline, the sequence and time ratios of signals on the mainline with different types of interaction and also the electric characteristics of the BIS receiving and transmitting circuits have been determined.

The proposed interface is oriented primarily toward design of single-board microprocessor systems or parts of them based on BIS with combined address and data buses and with levels of input and output signals compatible with the TTL levels. Functionally complete devices (processors, memory, controllers, input-output devices and so on), realized in the form of one or several BIS, are used as the components of the system.



Time Diagram of Mainline Control Transmission

Key:
1. Mainline request
2. Authorization for taking over mainline
3. Confirmation of request
4. Synchronization of exchange
5. Response of device

Table 1. Composition and Designation of Signal Lines of INTERBIS Interface

(1) Наименование линий и сигналов	(2) Буквенное обозначение	(3) Источник сигналов	(4) Приемник сигналов	(5) Назначение линий и сигналов
(6) *Линии передачи информации*	(8)		(10)	
Шина «адрес — данные» (7)	АД(00 : 15)	Ведущий (9) Ведомый	Ведомый Ведущий	Передача адресов и данных (11)
Линии управления обменом информацией (12)	(14)			(15)
Синхронизация обмена (13)	ОБМ	Ведущий	Ведомый	Указание начала и конца текущего цикла обмена
Прием адреса (16)	ПМА (17)	Ведомый	Ведущий	Указание завершения приема адреса ведомым устройством (18)
Ответ устройства (19)	ОТВ (20)	»	» (21)	Указание завершения приема либо выдачи данных ведомым устройством
Чтение данных (22)	ДЧТ (23)	Ведущий	Ведомый (24)	Указание готовности ведущего устройства к приему данных и завершения приема
Запись данных (25)	ДЗП (26)	»	» (27)	Указание достоверности данных на шине АД
Выбор устройства (28)	ВУ (29)	»	» (30)	Указание установки на шине АД адреса из зоны 160000_8 — 177777_8
Признак «запись — байт» (31)	ПЗП (32)	»	» (33)	Указание формата передаваемых данных (байт — слово) и указание операции записи
Ошибка при обмене (34)	ОШВ (35)	Ведомый	Ведущий (36)	Указание возникновения ошибки при обмене
Линии передачи управления магистралью (37)				
Запрос магистрали (38)	ЗМ (39)	Активные (40) устройства	Арбитр (41)	Извещение арбитра о готовности активных устройств к обмену (42)
Разрешение на захват магистрали (43)	РЗМ (44)	Арбитр	Активные устройства (45)	Извещение активного устройства о возможности захвата магистрали (46)
Подтверждение запроса (47)	ПЗ (48)	Ведущий	Арбитр	Извещение арбитра о том, что активное устройство стало ведущим (49)
Линии прерываний (50)		(53)	(54)	
Запрос на прерывание (51)	ЗПР (52)	Устройства	Активные устройства	Указание наличия в системе одного или нескольких устройств, требующих обслуживания
Разрешение прерывания (55)	(56) ПРР	Ведущий	Ведомый	Указание готовности прерванного устройства к чтению вектора (57)
Прерывание по внешнему событию (58)	ПРС (59)	Устройства	Процессор (60)	Указание необходимости перехода процессора на процедуру обработки этого прерывания по фиксированному адресу (61)
Дополнительные линии (62)				
Установка (63)	УСТ (64)	Пульт (65)	Устройства	Установка всех устройств системы в определенное исходное состояние (66)
Авария источника питания (67)	АИП (68)	Блок питания (69)	Устройства	Указание выхода постоянного питающего напряжения за допустимые пределы (70)
Авария сетевого питания (71)	АСП (72)	То же (69)	Процессор (73)	Указание возможности выхода постоянного питающего напряжения за допустимые пределы вследствие нарушения сетевого питания (74)

Key:

1. Name of lines and signals
2. Letter notation
3. Signal source
4. Signal receiver
5. Designation of lines and signals
6. Data transmission lines
7. "Address-data" bus
8. AD (00:15)
9. Master
10. Slave

[Key continued on following page]

[Key continued from preceding page]

11. Address and data transmission
12. Data exchange control lines
13. Synchronization of exchange
14. OBM
15. Indication of beinning and end of current exchange cycle
16. Address reception
17. PMA
18. Indication of completion of address reception by slave device
19. Response of device
20. OTV
21. Indication of completion of reception or transmission of data by slave device
22. Data readout
23. DChT
24. Indication of readiness of master device to receive data and to complete reception
25. Data recording
26. DZP
27. Indication of data reliability on AD bus
28. Selection of device
29. VU
30. Indication of setting of address from zone 160000_8-177777_8 on AD bus
31. "Record-byte" feature
32. PZP
33. Indication of format of data to be transmitted (byte-word) and indication of record operation
34. Error in exchange
35. OShV
36. Indication of error during exchange
37. Mainline control transmission lines
38. Mainline request
39. ZM
40. Active devices
41. Arbiter
42. Informing arbiter about readiness of active devices for exchange
43. Authorization to take over mainline
44. RZM
45. Active devices
46. Informing active device on possibility of taking over mainlines
47. Requestion confirmation
48. PZ
49. Informing arbiter that active device has become master
50. Interrupt lines
51. Interrupt request
52. ZPR
53. Devices
54. Indication of presence of one or several devices requiring servicing in system

[Key continued on following page]

[Key continued from preceding page:

55. Interrupt authorization
56. PRR
57. Indication of readiness of interrupted device to read vector
58. Interrupt by external event
59. PRS
60. Indication of need to convert processor to procedure for processing this interruption according to fixed address
62. Supplementary lines
63. Setting
64. UST
65. Console
66. Setting all devices of system to specific initial state
67. Emergency of power supply source
68. AIP
69. Power supply unit
70. Indication of DC supply voltage going beyond permissible limits
71. Emergency of network power supply
72. ASP
73. Processor
74. Indication of capability of DC power supply voltage going beyond permissible limits due to disruption of network power supply

An intermodular parallel interface was taken as a basis according to OST 11 305.903-80 when developing the INTERBIS interface. The main distinguishing features of the INTERBIS interface are:

expansion of the composition of mainline signal lines;

variation of the interaction protocol of the devices in the mainline takeover algorithm;

absence of restriction on minimum time of the address being held by the master device;

capability of using devices with different and unlimited speed in reception and (or) recognition of an address from below in a single system;

independence from continuously developed technology and circuit engineering.

These differences do not introduce discernible difficulties when integrating the intraboard INTERBIS interface with the interboard interface according to OST 11 305.903-80.

Functional characteristics of the INTERBIS. Agreements on organization of communications within system. Communication between two devices connected to the mainline is organized, as in most known buses, on the "master-slave" principle. The basis of exchange is the asynchronous principle; therefore, the algorithm for exchange between devices is not dependent on their speed and

exchange between two devices occurs at a speed determined by the speed of the slower of the existing devices. Devices of two types--active and passive--may be included in the system.

An active device is one which can perform functions of the master device and can receive interrupt requests during data exchange. This does not exclude the capability of designing an active device that performs all or part of the functions of a passive device.

A passive device is one which can perform the functions of only a slave device and can also transmit interrupt requests during data exchange.

The composition and designation of mainline signal lines. The connecting mainline consists of 33 signal lines joined (by functional feature) into five separate groups (Table 1). The level of the logic unit of the signals on all lines is low.

The address reception signal line is supplementarily introduced in the mainline, which permits organization of complete asynchronous interaction between devices, including transmission, reception and recognition of the address.

Agreements on takeover of the mainline. All active devices of the system may participate in takeover of the mainline and the operations should be performed in the following sequence:

each active device ready to become the master in exchange over the mainline transmits a ZM [mainline request] signal;

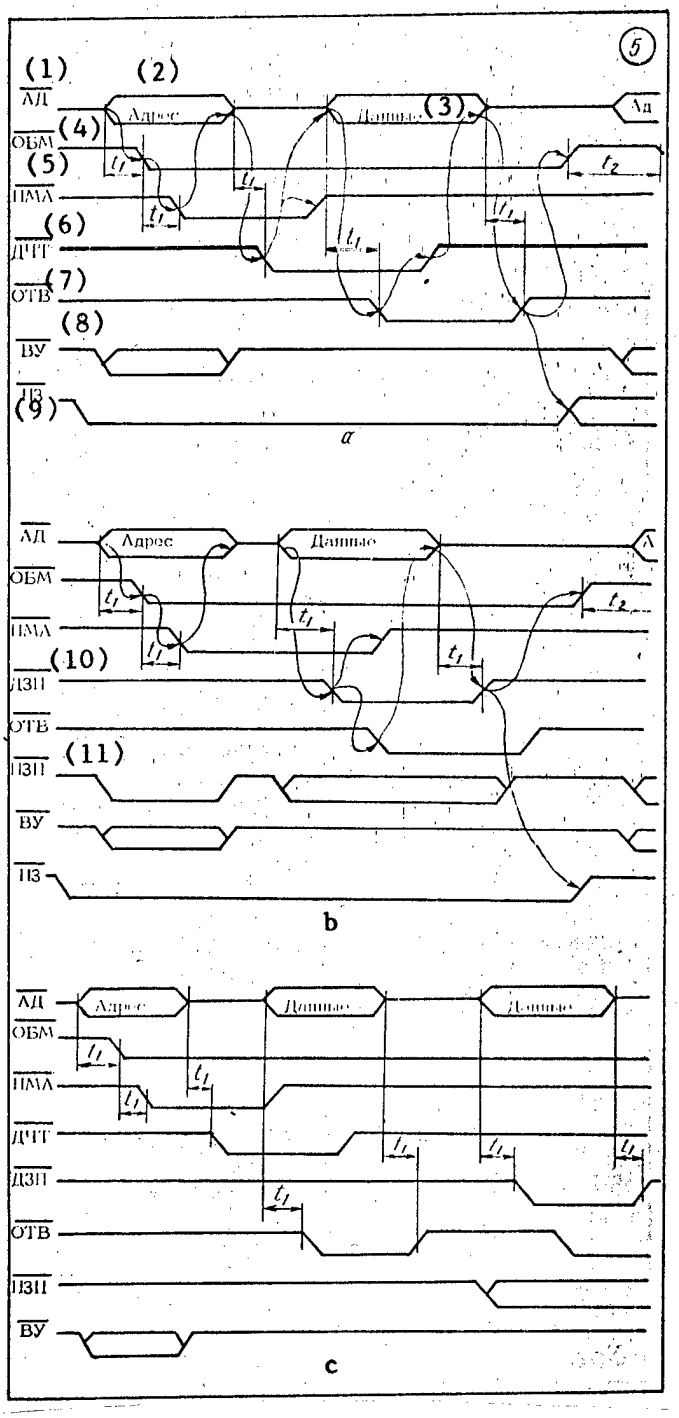
the mainline arbiter analyzes the status of the ZM line. If there is a ZM signal and if there is no PZ [confirmation of request] signal, the arbiter transmits an RZM [mainline takeover authorization] signal, which should be fed to the device with the highest priority to take over the mainline that has established the ZM signal;

the device that established the ZM signal and that has received the RZM signal sets the PZ signal if there is no OBM [synchronization of exchange] signal and triggers the ZM. The device that has established the PZ signal becomes the master;

the arbiter, having received the PZ signal, triggers the RZM signal. Resetting of the PZ signal is authorized to the master device only after the last of the OTV [response of device] signals in the given exchange cycle has been reset.

The time diagram of transfer of control to the mainline is presented in Figure 4. The priority of the devices is determined the same as in the interface according to OST 11 305.903-80.

Agreements on data exchange. Three types of data exchange operations--reading, writing and reading-modification-writing--are determined by the standard. The time diagrams of these operations are presented in Figure 5.



Time Diagrams of Exchange Operations: a--reading; b--writing; c--reading-modification-writing (t_1 = 0 min; t_2 = 25 ns)

Key:

1. Address-data bus
2. Address
3. Data
4. Synchronization of exchange

[Key continued on following page]

[Key continued from preceding page]

5. Address reception
6. Data readout
7. Response of device
8. Selection of device
9. Confirmation of request
10. Data recording
11. Record-byte feature

The maximum access cycle time is limited to 10 μs by this standard. If the access cycle to the slave device is not completed within 10 μs, then the master device frees the mainline. If the processor is the master, it converts to the standard interrupt processing procedure. If any other active device is the master, then it ceases exchange and informs the processor of the interruption that has occurred.

The PZP [record-byte feature] in the data recording phases can have both a low and high level, thus determining recording of a byte of any 16-digit word, and recording of a byte corresponds to the active level of this signal.

It should be noted that specific time ratios between signals are not established by the standard--their strict priority is guaranteed by introduction of time intervals of 0 minutes. This permits the use of the proposed interface for a high-speed BIS, whereas the minimum exchange cycle time is limited to 750 ns when using the interface according to OST 11 305.903-80.

Agreements on interruption. Any device in the system can emit interrupt request signals. In this case the interrupted device performs the following operations:

having detected an interrupt request (ZPR signal) on its own line, it performs the procedure of takeover of the mainline;

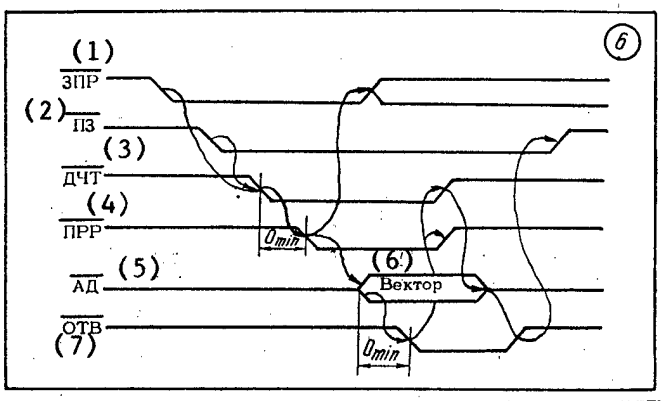
having received control of the mainline, it transmits PPR (sic) [interrupt authorization] and DChT [data readout] signals and reads the interrupt vector from the interrupt source;

it converts to processing of the interruption according to the vector.

The time diagram of vector readout during interruption is presented in Figure 6.

Electric characteristics of the INTERBIS. The electric characteristics for the receiving and transmitting circuits of the BIS (receivers and transmitters) have been established on the assumption that the length of the communications between BIS does not exceed 0.5 meter (this is quite adequate when the BIS are installed within a single board) and if the level of static noise does not exceed 0.4 V.

General requirements. The electric characteristics of receivers and transmitters should not depend on the characteristics of the BIS manufacturing technology. Transmitters and receivers should be compatible with each other on logic signal levels (with noise stability reserve not below 0.4 V) and on



Time Diagram of Interrupt Vector Readout

Key:

1. Interrupt request
2. Request confirmation
3. Data readout
4. Interrupt authorization
5. Address-data bus
6. Vector
7. Response of device

levels of current (output for transmitters and input for receivers). The ratio between the logic and electric states of the signals are presented in Table 2.

Table 2. Ratio Between Logic and Electric States of Signals

Логическое состояние (1)	Величина логического сигнала (2)	
	без черты над буквенным обозначением (3)	с чертой над буквенным обозначением (4)
0	не более 0,8 В (низкий уровень) (5)	не менее 2,0 В (высокий уровень) (6)
1	не менее 2,0 В (высокий уровень)	не более 0,8 В (низкий уровень)

Key:

1. Logic state
2. Value of logic signal
3. Without bar above letter notation
4. With bar above letter notation
5. Not more than 0.8 V (low level)
6. Not less than 2.0 V (high level)

Requirements on transmitters. Three types of transmitter outputs--standard output, output with open collector (channel) and output with three states--have been determined by the standard.

The transmitter should provide a low-level output voltage of not more than 0.4 V with inflow load current up to 3.2 mA, while the high-level output voltage

should not be less than 2.4 V with outflow load current up to 0.1 mA. The spurious capacitance from the transmitter output should not exceed 15 pF. The transition time from the logic zero to the logic one state and the transition time from the logic one to the logic zero state of the transmitter, presented to the receiver input, should be as short as possible and should not exceed 25 ns if the transmission line is 0.5 meter long and if the nominal capacitor load is 100 pF.

Requirements on receivers. The low-level input threshold voltage should not be less than 0.8 V with outflow input current of not more than 0.25 mA. The high-level input threshold voltage should not be more than 2.0 V with inflow input current of not more than 0.04 mA. The spurious capacitance from the receiver input should not exceed 10 pF.

Conclusions. The proposed INTERBIS intraboard interface has rather wide functional capabilities, permits microprocessor systems of different configurations to be developed, is compatible with the intermodular parallel interface according to OST 11.305.903-80 and can serve as the basis for development of a microprocessor set of BIS of increased functional complexity.

BIBLIOGRAPHY

1. Smolov, V. B., V. V. Barashenkov, V. D. Baykov et al, "Spetsializirovannyye TsVM" [Specialized Digital Computers], Moscow, Vysshaya shkola, 1981.

2. Hilburn, J. and P. Julich, "MikroEVM in mikroprotsessory" [Microcomputers and Microprocessors], Moscow, Mir, 1979.

3. Borshchenko, Yu. I., V. L. Dshkhunyan, Yu. L. Otrokhov et al, "The Elektronika NTs-80-01 Single-Board Computer], ELEKTRONNAYA PROMYSHLENNOST', No 1, 1981.

4. Fors, Zh., "A Standard Microprocessor Bus that Simplifies the Tasks of Microcomputer Developers," ELEKTRONIKA, Vol 51, No 15, 1978.

5. Ogdin, C. A., "Microcomputer Bus Standardization Makes (Dollars and) Sense," EDN, Vol 23, No 5, 1978.

6. Elmquist, K. A., H. Fullmer, D. B. Gustavson and G. Morrow, "Standard Specification for S-100 Bus Interface Devices," COMPUTER, Vol 12, No 7, 1979.

7. Digital Microcomputer Handbook, Maynard: Digital Equipment Corporation, 1976.

8. Balph, T. and I. Kister, "Microprocessor Bus Gears Up to a 32-Bit Future," ELECTRONIC DESIGN, Vol 28, No 14, 1980.

9. Yencharis, L., "But the Right Bus Evokes the System's Best," ELECTRONIC DESIGN, Vol 28, No 10, 1980.

10. Cushmann, R. H., "Microcomputer Bus Standardization--Will Designers' Needs be Met?" EDN, Vol 23, No 4, 1978.

11. Microprocessor Data Manual, ELECTRONIC DESIGN, Vol 26, No 21, 1978.

13. Alekseyevskiy, M. A., M. N. Gal'perin and V. O. Ignat'yev, "A Method of Investigating the Operating Modes of a Control Computer in Real Time," in "Tsifrovaya vychislitel'naya tekhnika i programmirovaniye" [Digital Computer Equipment and Programming], Moscow, Sovetskaya radio, 1968.

6521
CSO: 1863/72

COMPUTER TECHNOLOGY AND AUTOMATION AS THE BASIS FOR INTENSIFICATION OF THE NATIONAL ECONOMY

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 3-6

[Article by Doctor of Technical Sciences Vladimir Aleksandrovich Myasnikov, USSR State Committee for Science and Technology, Moscow]

[Text] The 26th CPSU Congress determined a need to convert the national economy of the USSR toward a primarily intensive path of development, which will permit fuller use of the country's scientific and technical potential and on this basis will make it possible to conserve all types of resources used for social production. The most important condition for raising the level of intensification is extensive use of data processing, transmission and display devices and efficient use of them in automatic and automated control systems of different problem orientation and different hierarchical levels.

The tasks posed by the 26th CPSU Congress in this field have been solved since 1981 within the framework of 12 scientific and technical programs, including four specific complex programs that have received priority with respect to support of the work with resources, since the main goals of this program should be achieved during the 11th Five-Year Plan. It is planned to complete fulfillment of the basic tasks of the specific complex programs through development of "turnkey" automatic and automated control systems by enterprises, organizations and institutions of various designation with guaranteed efficiency and organization of serial production of computer equipment.

Development of the Unified Program-Compatible Computer System (YeS EVM) in the 1970s in the country laid the basis for industrial development of computer equipment and software and made it possible to forecast the development of computer equipment and to guarantee the most rapid introduction into the national economy.

The main result of work on the SM EVM [International Small Computer System] was to create two families of minicomputers with standardized peripheral devices and software that meet the needs of different sectors of the national economy and that are designed primarily for use in ASU TP [Automated production process control system], SAPR [Automated design system] and ASNI [Automated system for scientific research]. The principles of the problem orientation of SM EVM have been worked out (mainly by specialists of the Institute

of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences), that take into account the specific technology of data processing in the systems and provide a significant reduction of expenditures and time for development and introduction of systems.

One of the achievements of the 10th Five-Year Plan is development and organization of serial production of microprocessors and microcomputers, which will permit the introduction of computer equipment in those sectors of the national economy where considerable capital expenditures were previously required for this. A total of 15 types of microprocessor sets with processing speed from 20 to 0.1 s is now being produced in the country, on the basis of which approximately 30 types of microcomputers, controllers and devices have been developed. The production of control microcomputers will increase approximately twentyfold during the 11th Five-Year Plan.

El'brus multiprocessor computer complexes, which laid the basis for development of new configurations of computer systems, were developed to solve complex scientific and technical and planning and economic problems that require a large volume of calculations. Work is also being completed on development of an essentially new multiprocessor computer with macroconveyor organization of calculations.

The situation in the country has been improved considerably with respect to the peripheral equipment of computers. The nomenclature of peripheral devices of the YeS EVM now numbers approximately 180 products and that of the SM EVM numbers more than 170 products (with regard to the equipment produced in CEMA member countries) and includes data preparation, input-output and storage devices. They include disk units with capacity of 100 and 200 Mbyte, tape units with high recording density, graphic and alphanumeric displays, multiconsole alphanumeric data display systems, various types of sequential and parallel printers, drum and flatbed plotters, data preparation devices based on magnetic carriers and other equipment. The development of operator-computer graphic interaction devices, which permits operational solution of problems of design automation and complex automation of scientific research, is of important significance.

The new peripheral equipment increases the efficiency of computers and of computer systems by increasing the data input-output speed 2-5-fold. Replacing punch cards and papertape with magnetic data carriers reduces the cost of data preparation by 30-50 percent, increases the labor productivity of computer operators by 50-60 percent, reduces the cost of information storage 1/8th to 1/10th and reduces the consumption of data carriers.

Extensive work has also been carried out in software development. Operating systems that provide processing in the packet and dialogue modes and in the multiprogramming mode have been developed for the YeS EVM. Program packs have been developed that expand the functions of operating systems, including the "Real-time supervisor," "Time-sharing mode" for computer multiplexing, the KAMA data remote control system. An operating system for the El'brus multiprocessor complex and a number of tape and disk operating systems for the SM EVM have been developed. Work is being conducted to develop the software for

distributed multimachine complexes, for VTsKP [Multiple-user computer center] and computer networks based on the second and third generation YeS EVM and also to develop YeS EVM-SM EVM complexing equipment. The OKA, INES, SIOD-ZOS, BANK-OS and SETOR database control systems and the POISK and ASPID-Z data retrieval systems and a number of others have been developed and recommended for widespread use. Introduction of these systems will make it possible to accelerate development of ASU [Automated control system] and of other data processing systems and will also guarantee interaction between various types of automated systems and will increase the computer load factor.

The appearance of microprocessors and microcomputers advanced the serious problem of developing software related to the wide variety of microprocessors oriented toward different applications. Cross-programming systems and programming automation systems are now employed as microcomputer programming devices.

Whereas cross-programming systems are available for practically all Soviet microprocessors, more or less satisfactory programming automation systems have been developed for only the microprocessor of the K580 series and those based on the SM-1800 and Elektronika K-1-10 microcomputers.

A specific unified program is now being developed which will make it possible to combine the efforts of developers to overcome the difficulties arising in this problem, to increase the reliability of programs for microcomputers and to reduce the cost of their production significantly.

Such specialized production complexes and other programming automation equipment as the RTK, PRIZ and TKP, which increase the labor productivity of programmers twofold or more, were developed during the 10th Five-Year Plan and recommended for introduction. Investigations must subsequently be organized to develop universal multilanguage programming systems that guarantee automation of all phases of program development and operation.

The State Algorithm and Program Fund (gosFAP), consisting of specialized sector and territorial funds, is expanding its activity from year to year. This source of obtaining software will have ever greater significance in the operation of computer systems as the software industry develops.

Extensive use of microprocessor equipment results in serious technical and social consequences, related to a change in the nature of teaching people, a reduction of manual labor and a decrease of the number of workers in industry, commerce and administrative services. Microprocessors will find application in the 11th Five-Year Plan in more than 200,000 different types of industrial and domestic devices and installations, while they will be found in 1.8 million facilities in the 12th Five-Year Plan, which is in itself a technical revolution. Machines and devices equipped with microprocessors have improved technical and economic indicators and new functional capabilities.

The use of microprocessors instead of ordinary computer equipment in existing or planned automated production complexes is very efficient. Thus, the use of the SM-1800 microcomputer instead of the widely employed SM-4 minicomputer

permits a more than tenfold increase of the mean cycles between failures and cost reduction of 1/7th to 1/20th, reduction of consumed power several tenths and a 1/20th to 1/50th reduction of the area occupied by computers . The saving calculated per system will comprise 150,000-200,000 rubles.

The problem of developing cooperation in the field of microprocessor equipment was one of the main problems at the 36th session of the Council for Mutual Economic Assistance, which was held on 8-10 June 1982 in Budapest.

Giving special significance to joining of efforts in the most important directions of scientific and technical progress, the heads of the delegations from the CEMA member countries signed general agreements during the course of the conference on cooperation and development and extensive use of microprocessor equipment in the national economy and also in development and organization of specialized and cooperative production of industrial robots. The program of work in microprocessors, approved by the session, and the coordinated measures of cooperation in the field of robot technology envision joint scientific research and planning-design developments, organization of specialized serial production of automatic production complexes of machines, devices and control systems equipped with the latest electronic devices.

A multilateral intergovernmental agreement was also signed on questions of specialization and cooperation in production of products for the microelectronics component base for computer equipment and materials and production equipment for microelectronics.

Realization of these documents will make it possible to strengthen the base for technical re-equipping of the national economy of CEMA members, automation of production processes in industry and agriculture and of many types of work in the control sphere, will make it possible to reduce the use of manual labor, especially in laborious and heavy operations, and will conserve energy, raw materials and materials.

The basic directions of scientific research work in the field of ASU TP, the number of which will double with each five-year plan, are an increase in the efficiency of systems by increasing their scientific and technical level and by reducing the labor expenditures for design of them.

The experience of automation of production processes and plants shows that production equipment and its control system should be developed as a unified automatic (or automated) production complex.

Accumulation of experience in automation of production and the appearance of microprocessor equipment have created prerequisites for design of automatic (automated) shops and enterprises without human and with flexibly adjustable production. To do this, a number of important scientific and technical problems must be solved, the main ones of which are:

a guarantee of the relability of equipment and control devices included in automatic production (besides the fact that high reliability indicators should be included in development of equipment components and diagnosis and

restoration of production equipment and control systems must be provided without interruption of the production process);

compatibility of units with each other and of production equipment with transport and loading devices and also with control devices;

a guarantee of the capability of flexible readjustment of production equipment from production of one type of product to another;

automation of auxiliary production management (preparation of tools and machine-tool accessories, removal of wastes, regeneration of some used auxiliary materials and so on);

automated preparation of production (calculation of production modes and preparation of control programs for design and production documentation).

Having begun to use computers in scientific calculations at the dawn of their appearance in the 1950s, scientists turned to development of powerful complex automated systems of scientific research--ASNI, which not only gather and process data during scientific experiments, but also help to interpret it by significantly reducing the path toward achievement of specific scientific results. More than 150 ASNI, which appreciably increased the scientific potential of the corresponding institutions, had been developed in the country by 1982.

The most important problem of modern design is the problem of achieving optimum solutions. These solutions can be found on the basis of comparison and analysis of tens and hundreds of complex versions, which cannot be carried out without resorting to computers. A means of intensification of the work of designers and planners has become automated design systems--SAPR. In evaluating their significance, one can say that we are talking about increasing the efficiency and quality of work of approximately 2.5 million persons, through whose knowledge and labor in achievement of scientific and technical progress, receive an "outlet" into the national economy.

The efficiency of SAPR is now quite significant, although the reserves for increasing it have been far from exhausted. For example, the SAPR at the LPO [Leningrad Production Association] Elektrosila permits a 12 percent reduction of the design deadlines, 6 percent conservation of electrotechnical materials by optimization of design and a 25 percent increase in the reliability of large electric machines.

During the 11th Five-Year Plan alone, 23 SAPR will be developed in organizations and at enterprises of industry, 41 SAPR and 16 production lines for design of construction facilities and 27 academic research SAPR in the country's vuzes will be developed from programs of the USSR GKNT.

Introduction of general-purpose computer capacities that guarantee development of new and development of existing ASUP [Automated enterprise management system] and OASU [Automated sector control system], will be doubled during the 11th Five-Year Plan. As is known, more than 4,100 production and scientific

production associations have been created and are operating in industry, which produce almost half (48 percent) of all industrial products. This high concentration of production determined the increase in the complexity of management, which made primary development of investigations in creation of ASU at these associations, the multiple scope of automation of all levels of management and transition from local to integrated systems objectively necessary.

Further improvement of the normative and management methodical materials is planned in creation and development of ASU for sectors, associations and enterprises that reflect modern advances in the field of the theory, methodology and practice of development and operation of ASU.

The basic directions of economic and social development of the USSR for 1981-1985 and for the period up to 1990 envision further development and increase of the efficiency of ASU and VTsKP networks and continuation of combining them into a unified statewide data gathering and processing system for accounting, planning and management of the national economy (OGAS).

Investigations are being conducted in further development of republic ASU, which are the main links of the OGAS, in the direction of developing intersector (planning, material and technical supply and so on) and sector (industry, agriculture, transport and so on) functional control complexes and automated data banks with extensive cooperation of scientific research and design work among developer organizations of republic ASU.

According to the hierarchy of ASU of different levels, their technical base--the state network of computer centers (GSVTs)--is also being constructed. According to the project, it will be a three-level system of computer centers that include up to 200 VTsKP at the upper level, up to 2,500 multiple-user VTs [Computer center] at the middle level and computer centers and user terminals of organizations and enterprises at the lower level. Development of 22 new and further development of six existing VTsKP is planned during the 11th Five-Year Plan.

The statewide data transmission network (OGSPD) should be a combination of means of communications and data transmission for a wide range of the country's enterprises and institutions. As indicated by foreign practice and investigations in our country, development of a data transmission network with pack switching is now the most promising in the composition of computer networks from the viewpoint of the efficiency of utilizing computer equipment.

Although much has been done in the country to organize wide use of computer equipment in all sectors of the national economy, there are still many organizational bottlenecks and unresolved problems which reduce the efficiency of using computer equipment. For example, more than 80 specialized scientific research, design and planning-production organizations are now involved in design and introduction of ASU and VTs alone. Moreover, there are structural subdivisions (departments, laboratories, sectors and so on) in 430 non-specialized organizations, which are developing various types of ASU. This results in unnecessary dissipation of forces and to excessive expenditures of monetary and material resources.

The USSR GKNT, jointly with the ministries, is conducting active work to eliminate parallelism and duplication in design of ASU and in guarantee of conditions for more efficient use of computer equipment in the sectors of the national economy. The most rapid solution of the now urgent problem of creating an information system with database on existing and developing software and also the problem of the economic and legal status of the gosFAP, specifically on relation of software to basic production products and of the pricing for software, will be of important organizational significance. A decision has also been prepared on exclusion of expenditures for information-computer investigations from maximum appropriations of control equipment, which will result in more extensive use of computers in the control apparatus and will increase its operating efficiency.

Far from all the problems related to an increase of the efficiency of use of computer equipment were touched on above. However, it is obvious that only complete automation of production, design and scientific research will permit us to fully solve the problems posed by the 26th Party Congress to increase the efficiency of social labor. And the decisive role in this is allocated to computer equipment and systems constructed on its basis.

6521
CSO: 1863/72

UDC 681.322

MICROPROCESSOR EQUIPMENT COMPLEX

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 9 Jul 82) pp 12-17

[Article by Corresponding Member of Ukrainian SSR Academy of Sciences Boris Nikolayevich Malinovskiy, Doctor of Technical Sciences Aleksandr Vasil'yevich Palagin, Candidate of Technical Sciences Valeriy Iosifovich Sigalov, Candidate of Technical Sciences Yuriy Sergeyevich Yakovlev, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev and Candidate of Technical Sciences Anatoliy Ivanovich Slobodyanyuk, Kiev]

[Text] Introduction. Soviet microprocessor technology is developing at rapid rates. More than 10 series of BIS [large integrated circuit] microprocessor complexes and several standardized series of microcomputers of different productivity, capable of satisfying a wide range of users, are now being produced. However, this striving results in the fact that the microprocessor equipment being developed and produced does not always correspond to specific applications.

Microprocessor devices of cybernetics technology,* which are problem-oriented control computer complexes based on BIS with regard to the specifics of industrial applications are more efficient. Their appearance is determined by the

*Cybernetics technology is understood as means of receiving, printing and processing of data in measuring, control, recognition and other engineering systems which usually operate in real time and which function without human participation.

The following features are typical for them: besides processing devices, the presence of devices for receiving data from an entity (process) and transmission of data to an entity (process), problem orientation or specialization of equipment and software to the class (type) of entities (processes) to be serviced, close interaction of processing devices with data input and output devices and in this regard, the mutual influence on their construction and design principles, requirements of maximum reliability, size and cost limitations and so on.

need to have effective, the least expensive, reliable and easy to use hardware in one or another class for automation of processes and entities.

Academician V. M. Glushkov, who advanced the idea of a general-purpose control computer at the end of the 1950s, which was realized with regard to a number of basic principles of cybernetics technology (problem orientation in the field of application, consideration of the mutual influence of the computer part and of the device for communicating the with entity on their design principles and so on) stood at the sources of cybernetics technology [1, 2].

Universal computer equipment was developed and is being developed primarily for automation of calculating operations.

The main development in them is creation of large collective-use universal computers, small computers for engineering and technical calculations and personal calculators.

Requirements of high productivity, convenience in servicing of large collectives and of individual users and simplicity of man-computer communication are placed on traditional universal computer equipment.

Unlike this type of computer equipment, the main thing in development of microprocessor devices of cybernetic equipment is the desire to automate the operation of various types of hardware, devices and systems to completely eliminate human labor in using them. Making calculations is not an end in itself here, but comprises part of the overall complex of work (mainly numerous logic operations) that are subject to automation. Moreover, the calculations usually have specifics determined by the features of specific applications.

The microprocessor devices of cybernetic equipment are supplemented by complex hardware for connection to entities during automation of them.

Thus, the computer equipment included in the microprocessor devices of cybernetic equipment has the following requirements placed on it:

algorithmic universality is limited by the range of the given class of applications;

the need for program and circuit orientation for specific classes of calculations, individual algorithms, frequently encountered logic problems and so on arises;

requirements on organization of the computing process vary and the main one becomes the requirement of real-time data processing;

different applications place their own restrictions on the design, reliability indicators and cost of computer equipment.

The software of microprocessor devices of cybernetic equipment also has its own characteristics, related to the use of standard programs and of internal and external languages oriented toward one or another areas of applications.

Moreover, one can note the increase of the role of "hard" programs, the use of a "truncated" or special operating system, cross-assemblers and simulation programs for preparation of machine programs and so on.

Microprocessor devices of cybernetic equipment can be used independently or jointly with minicomputers and universal computers. In the first case they serve to create local (autonomous) devices and systems for automation of different processes (data processing and transmission, measurements, monitoring and control, service and recognition and so on), not distinguished by high requirements on the data processing devices. In the second case they include hierarchical systems with minicomputers and universal computers.

Development and use of microprocessor devices of cybernetic equipment guarantee the following:

development of new structures and principles of design of various types of hardware and of systems, specifically in communications equipment, measuring equipment and so on;

bringing data processing devices directly to the points of producing it and conversion from centralized to distributed structures of data processing systems;

an increase of the "intellect" of devices, units and systems, i.e., an increase of their capabilities for data processing;

a high degree of parallellism in data processing, conversion to highly reliable structures of devices and systems with multiprocessor data processing;

reduction of the requirements on central data processing devices by increasing the requirements on local processing devices;

automation of functions that guarantee maximum convenience to users of hardware and systems;

automation of those processes and entities where the use of automation devices was previously impossible due to high cost;

acceleration of developments by conversion from schematic to program solutions using available means of automation.

The design of cybernetic equipment also requires a special approach. Thus, when developing oriented devices, one must employ available technology in the field of microprocessor equipment. One should also achieve the use of a unified design-production base in design of specialized digital devices that supplement universal devices.

They should contain general standardized units and standardized devices for connecting these units for rapid assembly of the necessary hardware. The nomenclature of the units of a series should be open, i.e., it should permit a

continuous increase, modernization and development of new types of units as needed, while design of them should be technologically simple and uniform that guarantees the capability of development and production of new functional units within short deadlines.

Development of specialized devices that supplement universal devices assumes careful analysis of the possible applications of cybernetic equipment and systems generalization of the requirements placed on them.

The first step on this path was development of a complex of microprocessor devices that include the MK-01 cassette microcomputer, the UVS-01 general-purpose microcomputer and the SO-01--SO-04 family of debugging systems designed on the basis of a modular set of microprocessor devices [3].

This set is a complete unit, structural and electrical organization and also design formulation of which permit configuration of microprocessor devices from them with different technical characteristics and functional capabilities.

Analysis of a wide range of applications of microprocessor devices and also of the current state of the Soviet component base made it possible to determine the requirements placed on the modular set of microprocessor devices of cybernetic equipment, universality, modular nature, interchangeability, functional completeness, capability of flexible change of configuration, capability of increasing the capacity of the set by development of new and modification of old modules and unified design solutions.

Structural organization and functioning of the complex. The MK-01 microcomputer [4] is designed for being built into measuring, monitoring-diagnostic and production equipment. This microcomputer includes the following modules:

a central processor based on the K580IK80 BIS;

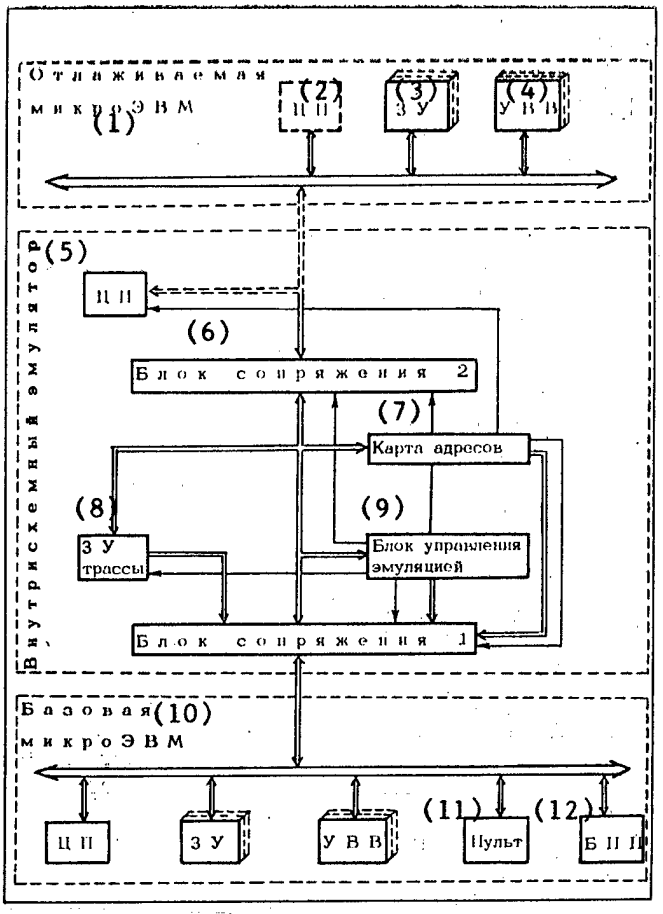
a main memory with capacity of 2 Kbyte;

three ROM modules of 2 Kbyte each, programmed once under laboratory conditions;

the MEK-488 collective-use line interface.

The design of the MK-01 is in the form of a cassette which includes six of the named modules that determine the functional capabilities of the microcomputer. A power supply with voltages of +12, +5 and -12 V must be provided for functioning of the microcomputer, the consumed power is not more than 17 W, the mass of the MK-01 does not exceed 1.5 kg and the dimensions are 212 X 150 X X 160 mm.

The UVS-01 microcomputer is designed for use in data measuring systems, automated communications sytems and special production equipment. It includes the following modules:



Key:

1. Microcomputer to be debugged
2. Central processor
3. Main memory
4. Input-output device
5. Intracircuit simulator
6. Integration unit
7. Address card
8. Memory routes
9. Simulation control unit
10. Baseline microcomputer
11. Console
12. Interrupt control unit

a central processor based on K580IK80 BIS;

two main memory modules with 32 Kbyte each;

a ROM with capacity of 4 Kbyte;

an interrupt processing module based on K589IK14 IS [integrated circuit];

a multiple-user line interface;

module for communication with standard peripheral devices (FS-1501 photo-reader, PL-150 perforator, Konsul-260.1 typewriter, DZM-180 ATs PU [alpha-numeric printer] and Simvol display);

integration with the control console;

power supply.

The UVS-01 microcomputer is located in an instrument housing measuring 550 X X 160 X 480 mm, the mass of the microcomputer does not exceed 18 kg and the consumed power is not more than 200 W.

The important components of the considered microprocessor complex, which guarantee latitude of introduction of the MK-01 microcomputer, is the SO-01--SO-04 system, designed for automation of design and debugging of software and hardware with respect to specialized devices constructed on a base of K580IK80 BIS. All models of the systems are based on the UVS-01 microcomputer and are distinguished by sets of peripheral devices and systems software.

The three younger models of the systems (SO-01--SO-03) are single-processor resident debugging systems. The common nature of bus organization of MK-01 and UVS-01 microcomputers and also some design features of the modules and a powerful engineering console of the modules permit rather efficient use of the system not only in autonomous debugging of software of microprocessor devices under development, but also in complex debugging of devices designed on the basis of the MK-01 microcomputer.

At the same time, the deficiencies inherent to all single-processor resident debugging systems are typical for the named models of the systems:

the programs to be debugged and the debugging programs are located in the same address field, which reduces the memory resources and input-output ports used by the device to be debugged;

the effect of the programs to be debugged and the equipment for debugging programs up to failure of the latter is possible during debugging;

realization of debugging modes in real time and when the programs to be debugged are in the read-only memory is not provided.

The older model (SO-04) is free of the enumerated deficiencies. Along with the usual set of peripheral devices (including a floppy disk unit based on the YeS 5074 disk drive), it contains the UVE-01 intracircuit emulator [5], which increases the efficiency of using the system in all phases of debugging of microprocessor devices.

The structure of the SO-04 is presented in the figure. This system consists of a baseline microcomputer, which includes the following set of units: a central processor (TsP), memory (ZU), set of standard external devices (UVV [input-output device]), control console and interrupt control unit, connected by a systems mainline. An intracircuit emulator (UVE) is connected to this mainline through integration unit 1.

Organization of the UVE provides two versions of integration with the microcomputer to be debugged, which includes a TsP, ZU and UVV. In the first

version, the UVE is joined to the microcomputer to be debugged through the jack of the microcomputer, designed for installation of the microprocessor. It is obvious that the microprocessor functions of the microcomputer to be debugged are taken on in this case by the microprocessor of the UVE. In the second version, the integration unit of the UVE is integrated with the system mainline of the microcomputer to be debugged. In this case the program to be debugged is implemented by the microprocessor itself of the microcomputer to be debugged.

The UVE includes a route memory, address card, emulation control unit and emulation TsP.

The route memory stores data on the sequence of states of the mainline of the microcomputer to be debugged during the run of the program; the address card modifies the address codes; the emulation control unit controls the operation of the UVE according to the given program and the emulation TsP replaces the microprocessor with the microcomputer to be debugged with the corresponding version of connecting the UVE to it and during autonomous debugging of programs.

Let us consider the operation of the debugging system in the second (the more common) version of connecting the microprogram to be debugged to the UVE. The functional program of the microcomputer is initially loaded in the memory of the microcomputer to be debugged, while the initial state of the program is loaded into the central processor of the computer to be debugged. The UVE guarantees direct access of the TsPU of the baseline microcomputer to the units of the microcomputer to be debugged by switching in the appropriate manner integration units 1 and 2. The internal assemblies of the TsP of the microcomputer to be debugged are loaded by sequential feeding of instructions and codes to be loaded into its mainline.

The internal units of the UVE are then programmed. The internal memories of the address card and of the emulation control unit are loaded. The entire memory capacity addressible by the TsP of the microcomputer to be debugged is conditionally divided into segments of 4 Kbyte each when programming the address card. The following information is given for each segment to be addressed (determined by the corresponding number of most significant address digits) during programming: authorization of access to the segment, location of the segment (in the device or baseline microcomputer to be debugged) and the absolute address of the segment if it is located in the baseline microcomputer.

Similar information is given for the input-output ports.

When programming the emulation control unit, the "special" point addresses are given, i.e., the points of the program in which emulation is stopped, the status of the mainline of the microcomputer t be developed is stored in the route memory, storage of data in the route memory begins or stops and the oscillograph synchronization signal is transmitted.

Integration unit 1 guarantees access of the central processor to the internal mainline of the UVE during programming of the internal units of the UVE.

The central processor of the microcomputer to be debugged should then be started, which processes the program until one of the given stop conditions is completed during programming (the central processor gains access to the given memory cell or to the given input-output port, a given time interval passes, the route memory is filled, one of the stop addresses will be received, the central processor of the computer to be debugged is stopped for a specific time interval and so on).

During emulation, integration unit guarantees data transmission from the mainline of the microcomputer to be debugged to the internal mainline of the UVE. The most significant address digits, which give the number of the segment, are entered in the address card. One determines here, according to the given program, whether access to the segment (and if it is not authorized, the program is stopped).

If access to the segment is authorized and if the segment is located in the microcomputer to be debugged, then access is gained in the ordinary manner. If the segment is located in the baseline microcomputer, the address card blocks access to the memory of the microcomputer to be debugged, modifies the address displayed on the mainline of the microcomputer to be debugged by substituting the most significant address digits that determine the number of the segment, the code of the number of the corresponding segment in the memory of the baseline microcomputer and permits direct access to the memory of the baseline microcomputer by guaranteeing access to the systems mainline through integration unit 1.

The address card also operates in similar fashion upon access to the input-output ports.

The codes from the mainline of the device to be debugged are fed through the internal mainline of the UVE to the inputs of the emulation control unit, which compares them to the given addresses of the "special" points. Upon identification of the "special" point addresses, the emulation control unit transmits the corresponding control signals to similar units of the UVE.

Data is also fed from the internal mainline of the UVE to the route memory, where it is stored according to the signals of the emulation control unit. After emulation has stopped, the central processor of the baseline microcomputer identifies the reason the stop occurred and if necessary selects data from the memory and internal assemblies of the central processor of the computer to be debugged and also from the route memory. The route memory is of greatest interest, since it was achieved during real-time execution of the program.

When the UVE is joined to the microcomputer to be debugged through the jack for installation of the microprocessor, the UVE operates similar to that described, but all functions of the central processor of the microcomputer to be debugged are performed by the emulation central processor.

The described organization of the UVE permits the use of the SO-04 in the following cases:

during autonomous debugging of software. In this case the UVE is not joined to the microcomputer to be debugged. The address card is programmed so that the memory and external devices of the baseline microcomputer take on the memory and UVV functions of the microcomputer to be debugged. The emulation central processor performs the functions of the central processor of the microcomputer to be debugged;

when the UVE is joined to the microcomputer to be debugged through the jack for installation of the microprocessor. In this case, the address card is reprogrammed in the corresponding manner as the ZU and UVV units of the microcomputer to be debugged are connected. Debugging in this configuration can be continued until the memory and input-output ports of the baseline microcomputer are totally free of the ZU and UVV functions of the microcomputer to be debugged;

upon connection of the UVE to the systems bus of the microcomputer to be debugged. The microprocessor is installed in the central processor of the microcomputer to be debugged. The functioning of the microcomputer to be debugged is checked and additional debugging is carried out if required. Testing and repair of the devices designed on the basis of the microprocessor are possible in the given configuration.

The UVE-01 parameters are presented below. The parameters of the widely ICE-80 emulator of the Intel Company are also presented for comparison.

Parameters	So-04	ICE-80
Capacity of route memory (in program steps)	512	44
Word capacity of route memory (in bits)	48	32
Number of "special" points	1,024	2

We note that the SO-04 has considerable hardware differences from the Intelec-230 model. However, despite these differences, functioning of the MDS driver of the ICE-80 can be guaranteed without significant program modifications.

Systems software of the complex. The aggregate of interrelated programs designed for automation of program development processes, debuggging and execution of them comprises the systems software of the complex of microprocessor equipment. Depending on the hardware used in development of the programs, the systems software can be divided into two groups: resident and cross.

Resident programs are realized on the UVS-01 microcomputer and cross programs are realized on instrument computers (YeS EVM or SM EVM).

The resident software includes a wide range of systems programs [6].

One of the main programs of the SO-04 is the emulator program, which guarantees control of the intracircuit emulator during debugging of the microcomputer to be developed and its functional program. Debugging is carried out in the interactive mode by using instructions which the operator enters into the system from the console. Let us consider the main ones of them.

Emulation instructions:

GO converts the UVE to the emulation mode until stop conditions are completed;

STEP converts the UVE to the emulation mode until the given number of instructions is completed;

RANGE defines the zones of the instruction addresses for which the microprocessors registers should be printed when performing the GO and STEP instructions;

CALL emulates the user interrupt system.

The interrogation instructions are:

BASE sets the display;

DISPLAY guarantees display of the memory, internal units of the microprocessor, the route memory and so on;

CHANGE changes the contents of the memory and of the internal assemblies of the microprocessor and the UVV;

XFORM sets the address card;

SEARCH performs a search for the given data in the cell memory.

Auxiliary instructions are:

LOAD loads the symbol table and entity codes from the input devices;

SAVE retrieves the symbol table and entity codes to the output device;

EQUATE enters additional symbols and their meanings in the symbol table;

FILL fills the given space of the memory with a specific code;

MOVE moves the data from one section of the memory to another;

TIMEOUT authorizes or prohibits a quarter-second interrupt in the operation of the device to be debugged;

EXIT transfers control to the program monitor.

Cross software is a set of translators and simulating programs realized on different production (cross and instrument) computers [7].

Conclusions. The complex of microprocessor devices of cybernetic equipment described in the article has been assimilated in serial production. The experience of using it in ASU, data measuring systems and devices showed that it is an effective means of mass computerization of different sectors of the national economy.

BIBLIOGRAPHY

1. Glushkov, V. M., "Computers and Automation of Production Control," VESTNIK AN SSSR, No 4, 1962.

2. Glushkov, V. M. and B. N. Malinovskiy, "Automation of Production Processes Using Computer Equipment," in "Materialy nauchno-tekhnicheskogo soveshchaniya" [Proceedings of a Scientific and Technical Conference], Moscow TsBTI, 1962.

3. Palagin, A. V., A. I. Slobodyanyuk and V. V. Sadovskiy, "Modular Set of Microprocessor Equipment," SREDSTVA SVYAZI, No 1, 1981.

4. Sasyuk, N. M., N. A. Gayvoronskiy, V. P. Kobylyanskiy and Yu. F. Khimerik, "Microprocessor Equipment in Measuring Apparatus," Loc. cit.

5. Badashin, V. V., V. L. Leont'yev, V. I. Sigalov et al, "Design and Debugging System for Microcontroller Hardware and Software," Loc. cit.

6. Malinovskiy, B. N., A. I. Slobodyanyuk and S. D. Pogorelyy, "Systems Software of Microcomputers and of a Debugging System Based on the K580IK80 Microprocessor," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 3, 1982.

7. Slobodyanyuk, A. I., S. D. Pogorelyy and S. G. Vaysband, "A Cross System for Debugging the Programs of the K580IK80 Microprocessor in Small Computers," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 2, 1982.

6521
CSO: 1863/72

UDC 681.324

SOME CHARACTERISTICS OF CONFIGURATION OF SM-2M COMPLEXES

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 p 133

[Article by engineer Veniamin Petrovich Kazachenko, engineer Anatoliy Dmitriyevich Lisitsa and senior scientific associate Tat'yana Ivanovna Podvizhnaya, Scientific Research Institute of Control Computers, Severodonetsk]

[Text] The most significant change in the design of the SM-2M compared to the SM-2 is conversion to designs of the second-generation SM EVM [International Small Computer System]. Type E2 cards (233.4 X 220 mm) are used here instead of type B cars (350 X 355 mm), which considerably enhances the technological effectiveness of manufacture of the units and facilitates adjustment of them.

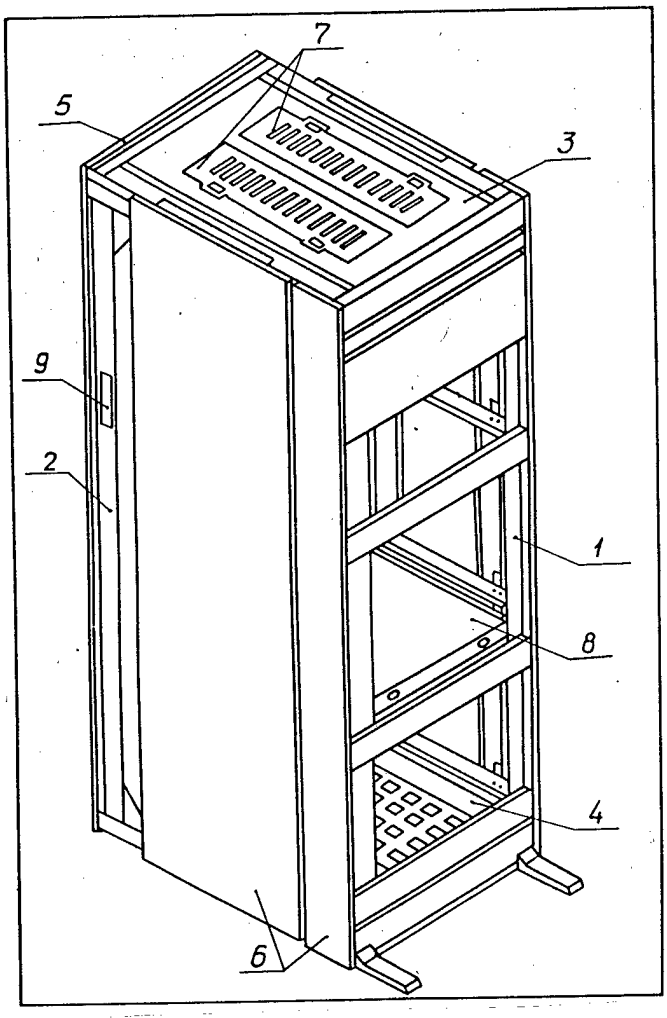
All the modules are made either in the form of a second-generation self-contained complete unit (AKB) or in the form of a block of components realized on a type E2 printed-circuit card installed in the AKB of another module, where a position, power supply and ventilation is provided for it. The characteristic feature of the AKB configuration in the SM-2M is the standard principle of arrangement of units in them, which are constituent parts of the AKB.

The processor and memory access channel are combined into a single module and are located in the same AKB. The processor, channel for communicating with the main memory devices and the input-output matchers use the same systems cables, due to which the number of systems bundles was reduced from 29 to 16 compared to the SM-2.

A second-generation cabinet with overall dimensions of 60 X 800 X 1,800 mm is used to configure the SM-2M complexes. The usable height of the cabinet was increased by 187 mm.

By increasing the useful volume of the cabinet and by reducing the number of modules in the central part of the complex, it was possible to reduce the number of cabinets in SM-2M complexes of maximum configuration, which enjoy a wide demand in the national economy.

Compact arrangement of the modules made it possible to reduce the length of the information bundles.



The cabinet is a structure consisting of two frames--a front 1 and rear 2, a cover 3 and base 4 secured by bolt connections. A door 5 is suspended from the rear side of the cabinet and the cases 6 are attached to the sidewalls of the cabinet. Two blower units 7 are installed on the hinged covers of the cabinet. An extensible table 8 is built into the cabinet for convenience of maintenance personnel.

Windows 9 with gaskets, which prevent dust from falling into the cabinet and which permit more efficient use of the cabinet blower system, are made in the side cases to separate communications between racks. This is of great significance since heat liberation per unit of cabinet volume has been increased in the SM-2M compared to the SM-2.

Installation of the plugs and front panels of the modules, which were earlier attached with screws that do not have heads that fall out, but are now supplied with fixers that permit them to be installed by light pressure, has been simplified in the SM-2M. This installation facilitates assembly of the cabinet and guarantees operating convenience to the user. Free movement of the

cabinet by installation of a wheel base in it is guaranteed. Special boxes that permit a considerable improvement of the aesthetic appearance of the product, that provide easy access to all the components of the devices (modules), including check points, and that guarantee easy communications between devices (modules) are used to space information communications between the central processors.

The bundles are reliably fixed when laid in the boxes, which reduces the load on releasable connections, eliminates bending and breaking of the circuit cards and increases the reliability of connection components.

The use of the new system of spacing the bundles resulted in a change in the design of the bundles. The disconnectplugs of the bundles are made only at angles and the bundles themselves have become more compact.

Power is supplied to the entire group of cabinets of the SM-2M complex by two cables through a three-channel distributor installed in the central cabinet. The main three-phase voltage from the distributing frame of 380/220 V is supplied by a power cable, while auxiliary power supply from a DC source with voltage of $\pm$ 300 V is supplied by the other cable (if needed).

Single-phase power supply of 220 V and power supply of +300 V are fed from each distributor channel by radial bundles to the cabinets so that the logic and electromechanical devices are connected to different phases.

The auxiliary power supply of +300 V guarantees efficiency of the logic devices during brief failures of the main power supply or when converting from the main power supply of the complex to a standby power supply.

6521
CSO: 1863/72

UDC 681.335.1

DEVICE FOR CONTROLLING CODING AND INPUT OF RANDOM PROCESSES IN A COMPUTER

Novosibirsk AVTOMETRIYA in Russian No 1, Jan-Feb 83 (manuscript received 7 Jan 82, final version 11 May 82) pp 28-30

[Article by M. S. Titov, "A Device for Controlling Coding and Input of Random Processes in a Computer"]

[Text] Digital processing of continuous random processes of a varying physical nature requires that each realization x(t) be coded by a finite sequence of discrete readings x_n, n=1 to N, and that the converted information be loaded in a computer memory [1]. The coding is done using A/D converters, and loading is generally done using a computer input device coupled with an A/D converter. Control of the coding and loading of information into the computer are very important when performing experimental research and solving applied digital processing problems. In particular, it is very important in selective coding to establish the moment of the beginning of coding and input of the chosen section of realization t_o, and the length of coding time $t_k=\Delta tN$, where Δt is the digitization interval. Note that the length of the sections chosen x(t) may vary, the order of sequence may be arbitrary, and the quantity may be significant.

Controlling coding and input by a computer program is not always efficient, frequently resulting in unjustifiably large expenditures of machine time. Known equipment methods do not provide sufficient automation of the digital processing process as a whole, or completely meet the wide range of demands made on input organization [1,2].

This article examines a device allowing almost completely automated organization of computer loading of random processes. The device's block diagram is given in Fig. 1.

The device contains a two-channel tape recorder; signal detection unit, consisting of a detector, low frequency detector and threshold device; switch; signal selection unit; pulse generator; AND and OR gates; time interval selection unit; coupling unit; commutator; fixed frequency tone generator; and A/D cpmverter coupled with the computer. The time interval selection unit is made with band-pass filters and a detector; the latter is connected with the

filters through the commutator. The fixed frequency tone generator is connected to the input of the tape recorder's second channel. The coupling unit is an ordinary thyristor key, which is also switched in through the commutator to the break in the feed circuit of the transistorized recording amplifier of the second tape recorder channel. As the commutator is used the ShI-17 step-by-step direct-action finder, having three contact bars, to which are connected the fixed frequency tone generator, band-pass filters of the time interval selection unit, and coupling unit, respectively [3].

This device provides automatic marking of the magnetic tape, such that on the second tape recorder channel are recorded the tone signals of fixed frequency f_i at time t_{0i}, i=1 to m; i.e., at the beginning of the ith realization sections to be coded and loaded into the computer.

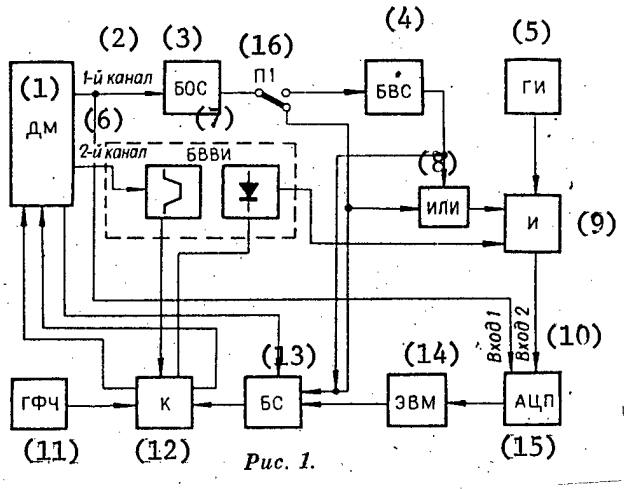


Рис. 1.

Fig. 1. Block Diagram of the Device

Key:

1. Two Channel Tape Recorder
2. First Channel
3. Signal Detection Unit
4. Signal Selection Unit
5. Pulse Generator
6. Second Channel
7. Time Interval Selection Unit
8. OR Gate
9. AND Gate
10. Input 1, Input 2
11. Fixed Frequency Tone Generator
12. Commutator
13. Coupling Unit
14. Computer
15. A/D Converter
16. Switch

The device operates as follows. The realization of x(t) is recorded on a circular magnetic tape in analog form, played back on the first tape recorder channel and fed to the input of the signal detection unit and to the signal input (input 1) of the A/D converter. There are no digitization signals at this time at the controlling input of the A/D converter (input 2); cording does

not occur. In the signal detection unit x(t) is detected, and its envelope extracted. Given pulses or individual overpeaks at the threshold device output, square pulses are formed that are fed through the OR gate to the input of the AND gate and the switchable input of the coupling unit. Signals from the pulse generator go to the other AND gate input. With selective input of one overpeak of x(t) from the series, the pulses go from the threshold device output to the signal selection unit input, which chooses the required controlling signal [3] and feeds it through the OR gate to the inputs of the AND gate and coupling unit. The signal selection unit is connected by the switch, and is used only in selective input of certain overpeaks. Continuous and sequential selective inputs are done without its participation. The coupling unit is used to switch in recording, using the commutator, on the second tape recorder channel; at the same time, from the output of the fixed frequency tone generator through the commutator to the second channel input is sent a tone signal of fixed frequency f_1, which is recorded on the second tape track. After the first revolution of the tape recorder ring, recording on the second tape recorder channel is turned off, and to the output of the second channel is connected the band-pass filter from the time interval selection unit, tuned to the f_1 frequency. Synchronous switching of the bars in the commutator is done by revolutions of the ring magnetic tape using a photorelay, which is switched into the feed circuit of the step-by-step finder and responds when a special marker hole in the tape passes between the illumination lamp and the photodiode. At the second ring revolution, there are signals at the AND gate inputs from the outputs of the signal detection unit (or signal selection unit), time interval selection unit and generator; at its output are shaped the digitization signals, which go to input 2 of the A/D converter, which codes x(t). Information is loaded from the A/D converter into the computer memory until a certain capacity is filled, after which the computer shapes a control signal that is fed to the switchable input of the coupling unit. This unit, through the commutator, again turns on recording on the second tape recorder channel; to the input of the second channel goes the tone signal of frequency f_2. At the moment when the third ring magnetic tape revolution begins, recording on the second tape recorder channel is turned off, and to the second channel output is connected the band-pass filter from the time interval selection unit, tuned to frequency f_2. Then, by the control signals from the outputs of the signal detection unit (or signal selection unit), time interval selection unit and pulse generator, at the output of the AND gate are shaped the digitization signals, which are fed to the controlling input of the A/D converter. Coding and input of the next x(t) section in the computer then occurs. Similar x(t) section coding and input cycles by ring revolutions continue until all the information to be digitally processed is loaded. The intervals between the inputs are used to preprocess the converted information and store it in peripheral memories of the computer.

This device can be used for coding and input of both instantaneous values of x(t) and values of its envelope A(t); they can be done simultaneously as well. The device's block diagram in this case is somewhat different, and takes the form shown in Fig. 2. To the device are added a second time interval selection unit, connected by a two-way link with the commutator, two switches (K1, K2), and second AND and OR gates. The A(t) envelope is taken from the low frequency

filter in the signal detection unit. The marking of the second ring magnetic tape track is done using the fixed frequency tone generator by frequency-separated signals of fixed frequencies f_i, f_i does not equal f . The x(t) sections are chosen by the second fixed interval selection unit and the second AND gate. Switches K1 and K2 are used for the feed to the A/D converter signal input at the specified moments x(t) or A(t) [4].

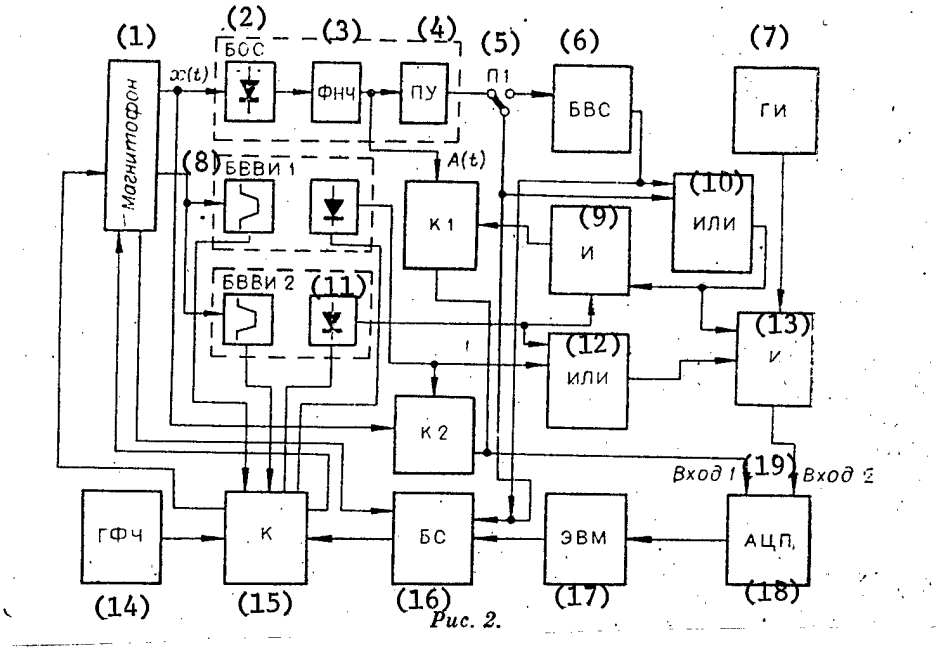


Рис. 2.

Fig. 2. Block Diagram of the Device

Key:

1. Tape Recorded
2. Signal Detection Unit
3. Low Frequency Filter
4. Threshold Device
5. Switch
6. Signal Selection Unit
7. Pulse Generator
8. Time Interval Selection Unit 1
9. AND gate
10. OR gate
11. Time Interval Selection Unit 2
12. OR gate
13. AND unit
14. Fixed Frequency Tone Generator
15. Commutator
16. Coupling Unit
17. Computer
18. A/D Converter
19. Input 1, Input 2

The device considered has been used to control coding and loading into the Mnsk-32 computer of random signals of the AF band, in solving various problems of digital processing of natural hydroacoustic information [2]. It was done using the MEZ-74 two-channel tape recorder, 2603 "Bruel & Kjaer" amplifier, G5-15 plus generator, KF-1 combination filter, serial UATsP2 coupled with a tape input device (UVvL). The input program (symbolic coding language) enabled a digital data set of variable length to be loaded. The converted information was recorded on the peripheral memory magnetic tape. The concrete length of the loaded set (t_k) was specified by the UVvL control word.

The device considered thus enables automatic marking of the magnetic tape by recording control signals on the second tape recorder channel, and simultaneous coding, input and preprocessing of information reproduced on its first channel. This provides selection of a substantial number of sections x(t) of arbitrary length (within the memory's free capacity), in any given order. Use of the device considerably increases information input and processing speed, and raises the accuracy of estimating random process characteristics.

BIBLIOGRAPHY

1. "Izmerenie i analiz sluchaynykh protsessov" [Measurement and Analysis of Random Signals], Dzh. Bendat and A. Pirsol, Moscow, Mir, 1974.

2. "Obrabotka gidroakusticheskoy informatsii na sudovykh TsVM" [Processing of Hydroacoustic Information on Ship Digital Computers], S. P. Rokotov, M. S. Titov, Leningrad, Sudostroenie, 1979.

3. Mironenko, M. V. and Titov, M. S., "A Device for Controlling Loading of Information into a Computer," (inventor's certificate No. 742 919), OIPOTZ, 1980, No 23.

4. Mironenko, M. V. and Titov, M. S., "A Device for Controlling Loading of Information into a Computer", (inventor's certificate No. 849 189), OIPOTZ, 1981, No 27.

9875
CSO: 1863/99

UDC 681.3.06+65.011.56

PAPERLESS PROGRAMMING TECHNOLOGY IN DIALOGUE ENVIRONMENT

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 20 Aug 82) pp 29-37

[Article by Doctor of Physicomathematical Sciences Igor' Vyacheslavovich Vel'bitskiy, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

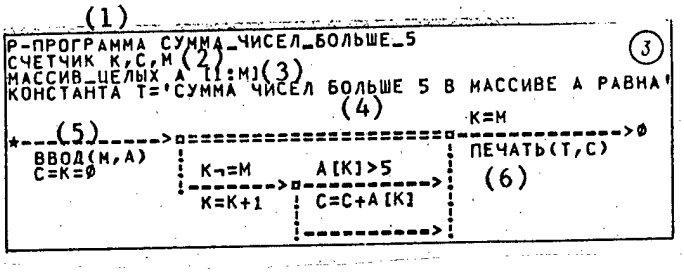
[Excerpt] Basic principles of P-technology. First principle. Unlike the traditional approach to programming, it is suggested in P-technology that the programs be drawn in the language of the loaded oriented graphs rather than written.

For example, the program for calculating the sum of all integers greater than five in file A of M integers has the form shown in Figure 1 in the language of P-technology. The condition for passage along the arc is written on top of the arc and the operation performed is written on the bottom of the arc in this figure, unconditional passage along the arc is noted by the symbol * if all the other conditions on the arcs corresponding to the given vertex are false. The corresponding program in the language of traditional block diagrams is presented in Figure 2 for comparison.

Unlike the language of block diagrams (graphs with loaded vertices), the language of P-technology is a graph with loaded arcs, which increases the clearness and technological effectiveness of using P-language in realization of it in modern input-output devices [9].

For example, writing this program in graphical P-language (on a display screen or an ATsPU [alphanumeric printer] has the form shown in Figure 3, where the arc without the arrow (of "=" symbols) is technologically effective and corresponds to identification of the vertices connected by it.

E. Deikstra arrived at the concept of stored instructions in his specific investigations on a rational programming discipline (technology) 7 years after first publications on P-language [10]. This concept fully coincides with that part of the P-language which is used to encode algorithms, but P-language is used over the entire technological range of program preparation (not only for encoding) and it is much clearer due to graphical writing of operators (stored instructions, according to E. Deikstra).



Writing of Algorithm in Graphic P-language on ATsPU

Key:

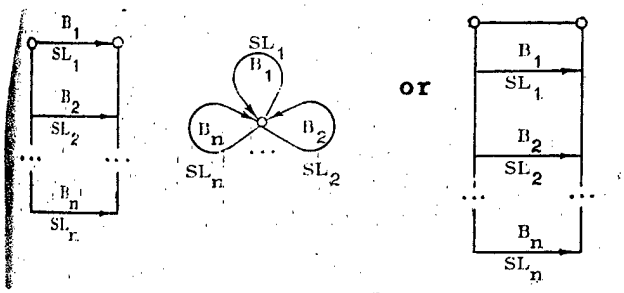
1. P-program: sum of integers greater than five
2. Counter K, S, M
3. File of integers A [1:M]
4. Constant T = 'sum of integers greater than five in file A is equal to'
5. Entry
6. Print (T, S)

For example, the main stored instructions [10] are written as:

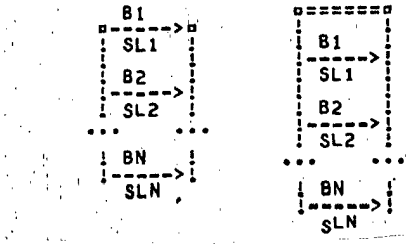
$$\underline{\text{if}}\ B_1 \to SL_1\ [] B_2 \to SL_2 [] \ldots [] B_n \to SL_n\ \underline{\text{fi}}$$

$$\underline{\text{do}}\ B_1 \to SL_1\ [] B_2 \to SL_2 [] \ldots [] B_n \to SL_n\ \underline{\text{od}}$$

The same thing in P-language has the form:



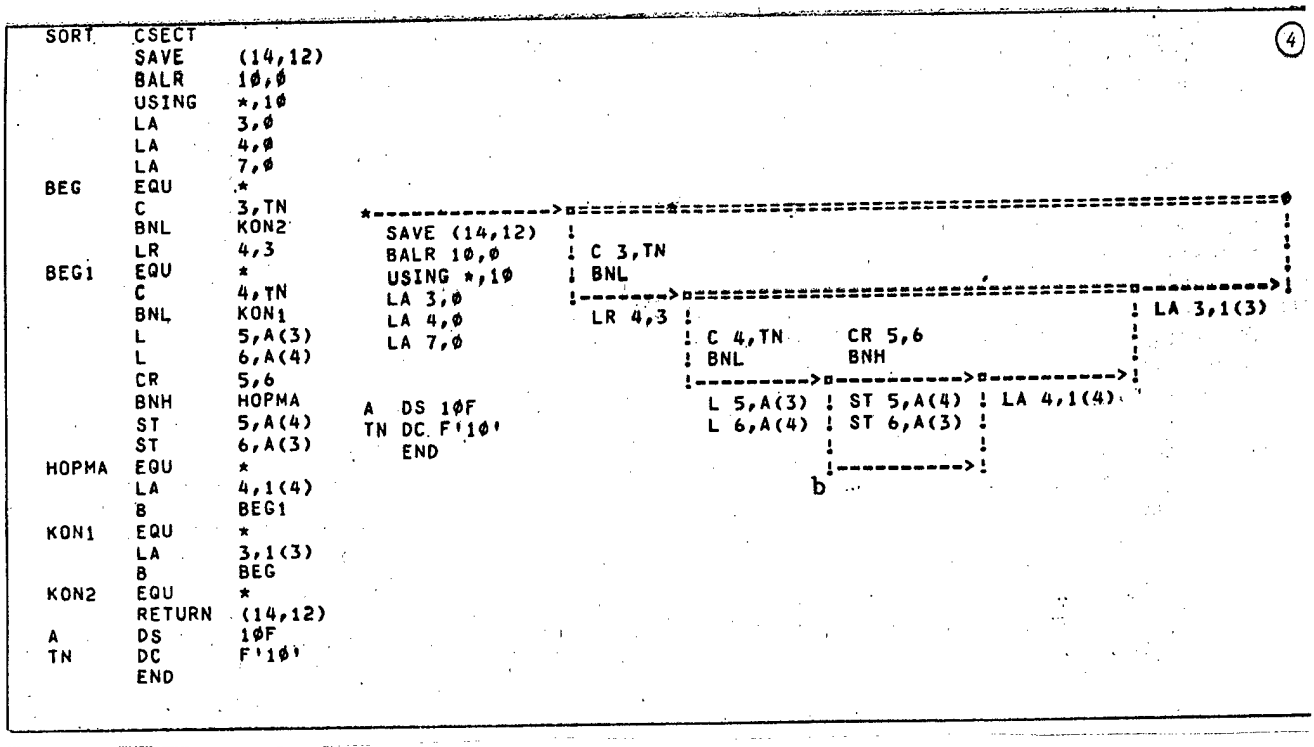
or in P-language represented on the input-output devices of modern computers:



The graphic P-language is some envelope into which any general-use programming language can be loaded: Assembler, FORTRAN, ALGOL and so on. Examples of writing programs in YeS EVM Assembler, FORTRAN and PASCAL in linear and graphical forms of writing on standard ATsPU, traditional for them, are presented in Figures 4, 5 and 6 for illustration. The indicated programs are converted to traditional linear form of writing by the corresponding translators of the RTK complex.

Consideration of the indicated examples shows that, first, the program in graphic P-language is clearer and better reflects ("at a glance") the structure of the algorithm: what follows what, what is embedded in what, under what condition, what is performed, where the branches and cycles are and so on; second, writing programs in graphic P-language is usually more compact (paper is conserved) due to more efficient use of the area of the display screen and page of the ATsPU; third, programs are entered in graphic P-language (due to the absence of marks, service words, use of a functional display keyboard and automatic drawing of arcs in it) faster--two-thirds fewer operations of the keys on the display is required.

Technologically, the program is stricter and more standardized due to automatic equalization of the inscriptions on the arcs and formation of the program from standard structures.



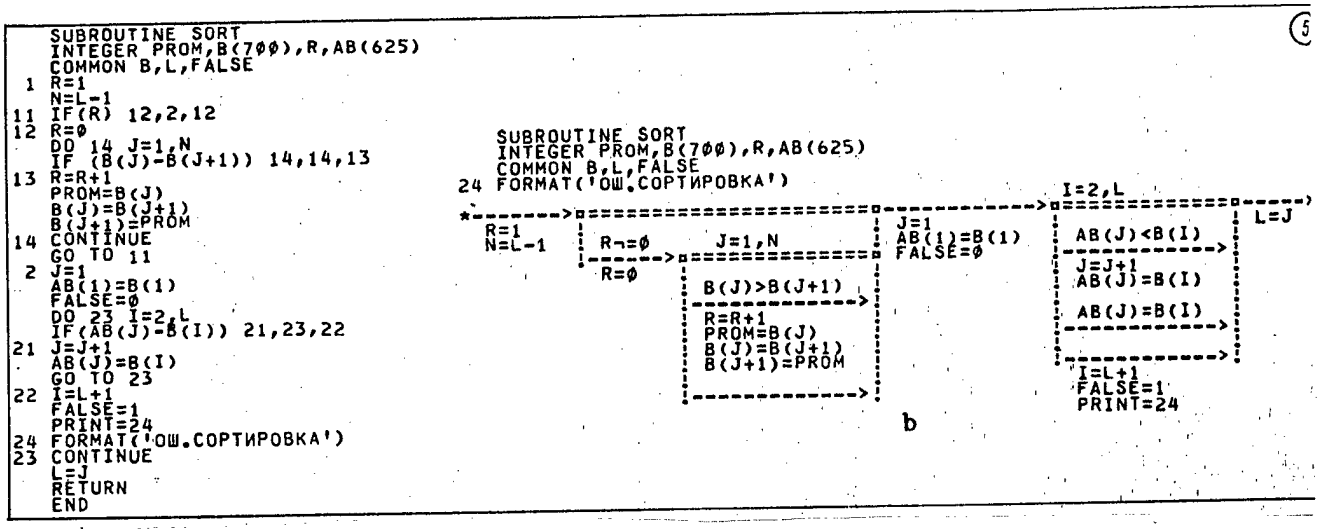
Example of Writing Program in YeS EVM Assembler in Traditional Linear (a) and Graphic (b) Forms of Standard Alphanumeric Printers

Graphical representation of the program is a more natural and convenient form of being given the algorithms for man. Conversion to the graphical form made it possible to introduce the concept of drawing and the related engineering discipline of managing the programming process and drawing, thoroughly developed in production practice, into programming.

The following three working formats of drawings: A3--ATsPU sheet (297 X 420 mm), A4--ATsPU half-sheet (210 X 297 mm) and A5--one-fourth ATsPU sheet (148 X X 210 mm)--are used in P-technology according to GOST 2.301-68.

An example of drawing a program formulated by P-technology in A-4 format is presented in Figure 7. Four fields are distinguished on the drawing.

The official field is the frame that bounds the drawing, the stamp in which it is indicated who made the drawing, when it was made, the decimal number of the project, the number of the order of the drawing and the name of the drawing, the number of the design level, the number of the version; the number of the previous page (drawing) is indicated on top, the number of the next page (drawing) on which the corresponding instruction (the arc of the R-graph) of the given drawing of the program project is decoded is indicated on each arc of the working field (the central part of the drawing separated by the symbol "-").



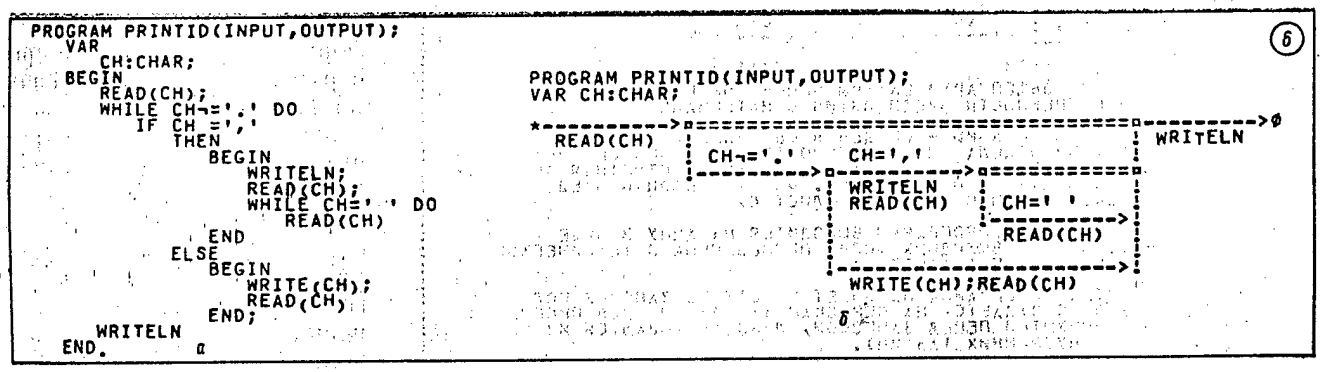
Example of Writing Program in FORTRAN in Traditional Linear (a) and Graphic (b) Forms on Standard Alphanumeric Printers

The specification field (the upper part of the drawing to the "-" symbol) contains a non-formal description of the goals and tasks of the given level (step) of design: designation, description of the algorithm, description of the input, output and transient variables and so on. It should be clear from the data indicated in the specification field (for the developer himself and users who will follow the corresponding program project) what the developer-program wanted to show in the working field of the given drawing.

The working field contains a formal description of the given design step in P-language. Loading on the arcs of the P-graph is arbitrary, without any restrictions whatsoever, i.e., any language (natural or machine), any alphabet (Slavic or Latin), the occupational language of any group of users (mathematicians, phsyicists, physicians and so on), can be used. The identifiers and notations on the arcs of the P-graph can be multilevel. If the arc of the graph (instruction of the P-machine) is decoded on the next design level, the number of the corresponding page (drawing) is indicated on the arc on the bottom in parentheses (see Figure 7).

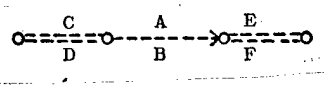
Variables and their type (real, integer and so on) can be described on top in the working field. Variables are described formally in one of the programming languages, determined at the stage of technological preparation. In the general case the working field of the drawing can be filled with text in any programming language and in the traditional linear form of writing it.

All the notations used in the working field are determined (formally in the programming language or informally in any language) in the field of abstractions (in the lower part of the drawing). To do this, the notations are automatically lighted up on the display screen in the field of abstractions. The developer should edit (remove, comment on or formally determine) the lines of the field of abstractions formed in this manner for subsequent accompaniment of them in the RTK.



Example of Writing Program in PASCAL in Traditional Linear (a) and Graphic (b) Forms on Standard Alphanumeric Printers

The second principle. Programming is carried out in P-technology from the following data: data are initially described in the language of graphs and the resulting graphs are then predetermined to the algorithms for processing these data. When describing in the language of graphs, the corresponding subject area is divided and documented into six areas in the working field of the drawing (it is divided into only three in HIPO-technology for comparison):



where A is the condition of fulfilling the instruction, B is the operations fulfilled by the instruction, C is the data at the instruction input and

coming from outside (from punch cards, from papertapes or from the communications channel), D is the data at the instruction input, stored inside the machine on an ML [tape unit], MD [disk unit], in the OZU [main memory] and so on, E is the data at the instruction output, transmitted from the computer to the ATsPU, papertape, punch cards or to the communications channel and F is the data at the output, stored inside the computer.

The data indicated on the technological arcs (denoted by the symbol "=") and the technological arcs themselves may be omitted.

The data in P-technology are described independently of their processing algorithms and the algorithms are found by predetermination of the graphs by only linear operators--of the type of conferring operators and functions written under the arcs of the graphs. Cycle operators and conditional and unconditional transitions are not used in predetermination--these operators are replaced by the data graph.

Only standard structures (P-instruction, complex of instructions, recurring complex of instructions and key) and two also standard procedures of the joining them (sequential connection and embedding in each other), accomplished automatically through the programmer's instructions [9], are used to describe the data and programs in P-technology. Because of this and the fact that the entire process of this description is accompanied by clear graphical representation, the programming process becomes especially reliable and does not contain most of the errors of traditional programming. It becomes possible to structure the working P-program so as to permit neutralization and sometimes even automatic correction of some types of errors in it that are undetected during debugging.

Programming from data in two phases sharply simplified development of programs in P-technology, since the skeleton of the algorithm is created during the first phase and the usually best known and easily formalized part of the problem--the data--is used for this and the more difficult operators and those of the cycle that cause most errors and operators of conditional and unconditional branches that confuse the program layout and technology of developing it are excluded from programming in the second phase.

The third principle. Programming in P-technology is carried out by the paperless scheme under the active control and monitoring of the computer. In this scheme, all documentation and all drawings of the design are stored in the computer memory on magnetic carriers. Only an annotation, title of the design and if necessary the first upper levels (drawings) of the tree of the design and instructions for starting the correcting system are issued to the user in hard copy (on paper). The user receives the remaining (incomparably large) part of the data about the program scheme in the automated mode on the display screen. The means of automation of the RTK complex easily permit this data to be found and to refine it when shifting upward (downward) along the tree of the scheme.

The computer forms and stores the hierarchy of the executors, defining their reasonable specialization, stores all obligations and commissions of each

programmer and the coordinated deadlines for completing the work, follows completion of these deadlines, gives timely information about disruption of them, records and stores all operations of the developers in the design archive, carries out communications between developers and passes on orders and instructions, adopted agreements and mutual obligations. Moreover, the computer blocks data against unsanctioned access, forms and stores the multilevel network of design frames, automatically collects design frames into the same working P-program and so on.

The greater part of the technological data in the paperless scheme is obtained automatically from the useful working data which circulates in the computer during preparation of the programs. All programmers are connected through their own display screens to some electronic mail network in which paper is used in exceptional cases.

The scenario of this organization is stipulated beforehand during the phase of technological preparation and reduces to preparation of the corresponding set of technological programs [8]. In those places where these programs do not exist in the production process, the computer asks the man for assistance, who works himself instead of the absent programs and enters the final result of his work in the computer at its request. This eliminates the dead end of the proposed technology and guarantees unlimited capability of phase automation of it.

Organization of work in P-technology. The presently known forms of organization of work in programmer collectives are based on the principle of "everything headed" by the supervisor, chief programmer and so on. This form is typical for the initial and low levels of organization of collectives.

Because of the characteristic of the graphic language used in P-technology, it became possible to create a higher hierarchical organization of work with deep specialization of executors. The two most difficult problems to traditional programming are solved in this case: parallelling and planning of the programmers' work.

Work is parallelled in the design phase, which is carried out from top to bottom* at levels in the language of loaded oriented graphs. Each arc of the graph of the higher design level is determined (interpreted and refined) either in the field of abstractions of the given drawing (design frame) or on the drawing (drawings) of subsequent levels.

This determination of each arc of the upper level graph can be entrusted to the corresponding collective of programmers and they in turn, by determining its components and writing them in the form of graphs, thus assign who and in what sequence they will be done and so on. Thus, the hierarchical scheme of the programmer collective is formed during determination of the upper levels of the corresponding program scheme by P-technology.

*It is assumed that the phase of technological preparation is carried out from bottom to top in P-technology, which guarantees the required high level of standardization and unification of design solutions during subsequent working design from top to bottom.

Writing the algorithms in natural language and in graph form, the use of data in the field of specifications and abstractions and also paperless operational access to it permit effective verification (expert analysis) of the design process and guarantee and high level of automation and reliability. Moreover, the process of communication of the customers, developers and users with each other is facilitated--elements of excess emotion in contacts are eliminated from it.

The work is carried out at an automated programmer position (one or several displays working in dialogue mode under control of the RTK complex). If there are several displays at the programmer position, then one of them can be the main (active) one with a keyboard, while the remaining ones are auxiliary displays. They operate in the monitor mode (without keyboard) or only in the mode of displaying data about previously adopted design solutions, input-output data, corresponding items of the technical assignment and so on. The auxiliary displays can be collectivized for the programmer collective in the same room or can be replaced by a single large semiscreen. The operating medium of the programmer collective and of each member of it is displayed on these screens.

The programming process is organized in P-technology by the machine-man-machine scheme with the active participation of the machine in formation and monitoring of the activity of man. The machine begins the process of designing the program scheme, assigning to man those questions and answers to which are required for correct organization of the production process. The questions and the flow sheet (line) themselves are formed by the programmer in the technological preparation phase as a result of generalization of his own experience and the experience of other collectives, including the useful experience reflected in the standards, and established in the starting version of the RTK complex.

Planning the work in P-technology and monitoring its execution are carried out by the network method. The real network schedule of work of the programmer collective is formulated automatically in the RTK complex from the operating data circulating in the computer and obtained when filling in the certificates of the executors, the working fields of the upper level drawings, current work personnel and so on [8].

The network schedule of work is formulated in some modification of the graphic P-language, in which the arcs emerging from one apex and simultaneously corresponding (parallel) to the work of the network schedule to be performed are connected by the symbol ":" instead of "!" in the graph programs. The loads on the arcs of the graph assign the number of the work by order and length in weeks: on the working arcs in parentheses and on technological arcs without parentheses. The technological arcs determine the time reserves of the executors to complete the corresponding work.

The path in the network P-schedule, not containing technological arcs (time reserves), is called the critical path. The work is decoded down the network P-schedule: the number in order, the name, the responsible executor and the calendar deadlines of the beginning and end of the work.

Relationship to programming languages. Existing programming languages do not meet the requirements of the technological effectiveness of their use. Almost all known algorithmic languages are rigidly determined--they are usually too artificial and complex for professional use, much effort must be expended on assimilation of them, but their disadvantages and deficiencies for specific application then become obvious almost immediately.

As our experience showed, the programming language in its modern interpretation determines (solves) only 10-15 percent of production problems over the entire range of development and operation of the programmed product. Modern programming languages are not intermediaries either between the developers of the program product or between its users or between the developers and users.

A program written in programming language is a "thing in itself," in which an enormous amount of information containing only the final results of extensive mental activity of the programmer, is enciphered and encoded in compressed form. During this activity, the programmer determines some abstract model of his own program, the model of the external medium (operational situation) in which it should operate, the characteristics and all the refinements of the interaction of these models, determines the selection of the rational method of program implementation, the sequence of phases and operations to prepare it and so on.

Unfortunately, all this information remains "behind the frame" in the programmer's head and interpretation of it by final writing of the program in modern algorithmic language requires considerable effort. This significantly complicates the technology of manufacture and operation of program products.

The main working (programming) language in P-technology is the corresponding natural language without restrictions. The working technological language of the programmer in P-technology is included in some format or envelope (graphic representation, writing above or below the arc, the field of specifications, the field of abstractions, the design frames, the order of filling them and so on), which determines the rational operations of the programmer collective in manufacture of the corresponding program product.

Programming languages, including the traditional widely used ones (FORTRAN, ALGOL and so on), serve in P-technology as only some "filler" of the corresponding production format (envelope) at a specific (small) phase of development of the program product--in the phase of encoding fragments of algorithms for operation of the computer. The envelope of the technological P-language and the form of using it over the entire range of development and operation of the program product are supported by the corresponding automation equipment--the RTK complexes. This equipment guarantees the clearness of programs required for computer operation and written by the developers in natural P-language.

The described concept of programming language and its role in the production process of the program article is contradictory in some sense to the presently advanced concept of a common programming language. From the viewpoint of the principles of P-technology, the concept of a common programming language

(of the ADA type) is now just as harmful as 20 years ago, when ALGOL-60 was touted no less intensively for the role of this language.

This is determined by the fact that the concept of a common language detracts from solution of real programming problems and aims at solution of imaginary and simple problems. The fact is that the programming language in its modern interpretation determines (solves) only 10-15 percent of the production problems; therefore, even a supergood new language does not solve the problem of rapid development of high-quality programs by the programmer collectives, as ALGOL-60, PL-1, ALGOL-68, PASCAL and so on (a total of more than 2,000 languages) have not solved it. There will also be no miracle with the ADA language.

The relationship of P-technology to the standards. The fact that the concept of the drawing has been introduced in P-technology and that the program is depicted in the form of a graph makes it effectively standardized according to the existing principles of YeSKD [unified system of design documentation], YeSTPP [unified system of technological production preparation] and so on.

The fact that the RTK complex is new in the apparatus of the technological frames introduced in it [8] fully automates the process of using the standards and monitoring their observance. The work is organized in such a manner that the machine determines the sequence of human operations in the production process of preparing the program design and it assigns questions to the programmer and indicates in difficult cases how to answer them correctly. The sequence of questions and the recommended content of the answers are included in the RTK complex during technological preparation. These questions and drafts of the answers are compiled by highly qualified programmers with regard to the collective experience reflected in existing standards.

Since the questions and answers are tied to a specific collective and to the problems to be solved by it, the programmer's work occurs in a natural manner only according to the existing recommendations of the standards and failure to observe them becomes impossible (the programmer cannot work otherwise). The computer in this technology takes on itself extensive rough work in formulation of technical documents from the programmers' answers according to the requirements of the standards.

Since P-technology is paperless, a new opportunity arises to configure the documentation hierarchically on magnetic carriers, and to automate the process of finding the necessary location of the documentation. This considerably simplifies the process of operation and accompaniment of program projects designed by P-technology.

Effect on computer configuration. It is known that the configuration of modern computers was formed gradually mainly on the basis of the engineering principle of achieving the maximum speed and reliability of the computer with minimum expenditures of the number of components, consumed power, occupied volume and so on.

However, the effective speed of a computer is now largely determined by the speed at which man works in compiling the programs for it. As a result, the

cost of programs exceeds several-fold the cost of the entire hardware and has a constant tendency to increase.

The principles of P-technology permit formulation of five basic requirements on computer configuration, oriented toward the programmer:

this computer should work with data in symbolic form and in the language of loaded oriented graphs;

the computer should contain its own internal alphabet assigned by the programmer himself at the phase of technological preparation for program production;

the memory in a P-machine, unlike that of a computer, consists of individual functional types: for storage and processing of tables, for storage and processing of large-dimension matrices, for gathering of symbolic data and so on. Both the nature of data storage and the permissible mechanisms of access and processing of it for the programmer are structurally determined in these memories;

completion of the program of the P-machine is directly dependent on the data to be processed and is controlled by the data;

the process of completing the program in the P-machine is parallelled each time with a factor equal to the number of arcs emerging from the completed apex of the P-program. This parallelling, like a chain reaction, can be applied to a large number of P-processors and can be implemented fully automatically without human participation. Because of this organization of the P-machine, programmers have the capability (essentially absent in modern programming) of converting to a simpler technology of designing indeterminate P-programs that permit simple parallelling of work in the collective and simple transition from informal models to formal descriptions of them.

The described computer configuration is especially effective for multiprocessor microcomputers with high degree of integration and specialization of individual processors with developed means of rapid communication between them.

BIBLIOGRAPHY

1. Glushkov, V. M., "Osnovy bezbumazhnoy informatiki" [Fundamentals of Paperless Informatics], Moscow, Nauka, 1982.

2. Glushkov, V. M., "Fundamental Research and Programming Technology," PROGRAMMIROVANIYE, No 32, 1980.

3. Glushkov, V. M. and I. V. Vel'bitskiy, "Programming Technology and Problems of Automating It," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1976.

4. Glushkov, V. M., I. V. Vel'bitskiy and A. A. Stogniy, "A Unified Automated Linguistic Approach to the Technology of Developing Applied Program

Packs," in "Struktura i organizatsiya paketov programm: Tezisy dokladov mezhdunarodnoy konferentsii" [The Structure and Organization of Program Packs: Report Topics of an International Conference], Tbilisi, Metsniyereba, 1976.

5. Glushkov, V. M., I. V. Vel'bitskiy and A. A. Stogniy, "One Approach to Design of the Systems Software of Modern Computers," KIBERNETIKA, No 2, 1972.

6. Glushkov, V. M., "The Theory of Automatons and Problems of Designing the Structures of Digital Computers," KIBERNETIKA, No 1, 1965.

7. Glushkov, V. M., "Sintez tsifrovykh avtomatov" [Design of Digital Automatons], Moscow, Fizmatgiz, 1962.

8. Taranenko, A. A., V. Yu. Kayurov, V. F. Kirsanov et al, "Organization of Work in Paperless Programming P-Technology," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1982.

9. Sholmov, L. I., V. N. Khodakovskiy, V. V. Makarov et al, "Graphical Representation of Algorithms in Paperless Programming P-Technology," Loc. cit.

10. Deikstra, E., "Distsiplina programmirovaniya" [Programming Discipline], Moscow, Mir, 1976.

6521
CSO: 1863/72

UDC 681.3.51.6.42

ORGANIZATION OF DESIGN PROCESS IN PAPERLESS PROGRAMMING P-TECHNOLOGY

Kiev UPRAVLYAYUSHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 20 Aug 82) pp 38-43

[Article by engineer Aleksandr Akimovich Taranenko, Candidate of Physicomathematical Sciences Vasiliy Yur'yevich Kayurov, engineer Viktor Fedorovich Kirsanov, Candidate of Technical Sciences Aleksandr L'vovich Kovalev and engineer Aleksandr Sergeyevich Kosyak, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Excerpts] Introduction. The productivity of the collective labor of programmers depends considerably on the efficiency of managing the program design. It is feasible to have not only traditional means of automation of direct program preparation (translators, text editors, debugging devices and so on), but also dialogue devices for supporting the design process and the organizational activity of the programmer collective from this viewpoint in instrument software systems. These devices are the core of the automated program production control system, in which, unlike most other types of ASU, there is the capability of fully automating fill-in of the database.

The importance of this paperless document circulation to support the efficiency of production management was emphasized by Academician V. M. Glushkov [1, 2] as one of the basic principles of ASU design.

Solution of problems of organizing the work of the programmer collective is closely related to the program design process, which guarantees joining of the informal description of the problem and its formal expression in terms of algorhythmic (machine) language [3, 4].

Program design is now carried out mainly by two methods. The first method assumes an increase of the level of formal descriptions so as to shift the boundary of formalization as high as possible. The second method is contradictory to some extent--the descriptions remain informal in essence, but nevertheless are processed by formal means and are provided with the corresponding instrumental support. Common among these directions is the desire to minimize routine operations, entrusting them to the instrumental support.

The program production support system proposed in the paper combines the advantages of these two directions. The system is realized as a further development of the RTK complex and is used primarily to support the operations performed by P-technology [5].

From the viewpoint of automating the management of program designs, the considered approach dictates a strictly defined production discipline to the programmer collective and management. The specifics of concrete designs can be taken into account be readjusting the system for the required composition and sequence of production operations. The task of the system includes automatic accounting of operations to be performed, detection (and possibly blocking) deviations from the regulated technology, monitoring of planning deadlines and also statistical data gathering and output required for correct planning and normalization of programmer labor.

Conclusions. The considered design support system is used during the entire program production cycle: from analysis of the initial task and compilation of the first informal specifications to printout of the documentation for the finished program product.

The production process is postulated so that the necessary technical documentation is prepared simultaneously with the main work related directly to program development. Automated output of final and intermediate documentation is accomplished by the procedures contained in the system according to the adopted requirements and standards.

All the data about the design is stored in a form accessible for finding references, which provides the management with reliable data on the status of the program design. Moreover, the system in this plan takes on an active role, informing of the approach or interruption of routine planning deadlines without special initialization.

The system makes available different services as a function of the phase of production and the category of users. Let us especially note its following features:

support of step detailing of data and algorithms in graph language;

support of debugging the program design, which is in different stages of preparation and formal determination;

automation of planning, distribution and monitoring of performing operations on program preparation in the programmer collective.

Control of the design program is characterized by a sharp reduction of the volume of paper documents circulating within the collective. In this case the system guarantees timely availability of the necessary data to developers, actually performing the role of "electronic mail."

All the work of the programmer collective on program design is subordinate to the unified production discipline for which the system is adjusted at the phase of technological preparation of program production.

The noted capabilities advantageously distinguish the proposed system from the known means of automation of program production.

BIBLIOGRAPHY

1. Glushkov, V. M., "Vvedeniye v ASU" [Introduction to Automated Control Systems], Kiev, Tekhnika, 1974.

2. Glushkov, V. M., "Osnovy bezbumazhnoy informatiki" [The Fundamentals of Paperless Informatics], Moscow, Nauka, 1982.

3. Glushkov, V. M. and I. V. Vel'bitskiy, "Programming Technology and Problems of Automating It," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1976.

4. Vel'bitskiy, I. V., "Paperless Programming Technology in a Dialogue Environment," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1982.

5. Vel'bitskiy, I. V., "P-Technology-80," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1980.

6. Sholmov, A. I., V. N. Khodakovskiy, V. V. Makarov et al, "Graphical Display of Algorithms in Paperless Programming P-Technology," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1982.

6521
CSO: 1863/72

UDC 681.3.056

GRAPHICAL DISPLAY OF ALGORITHMS IN PAPERLESS PROGRAMMING P-TECHNOLOGY

Kiev UPRAVLYAYUCHSHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82
(manuscript received 20 Aug 82) pp 43-47

[Article by Candidate of Technical Sciences Leonid Ivanovich Sholmov, engineer Vasiliy Nikolayevich Khodakovskiy, engineer Valeriy Vasil'yevich Makarov, engineer Oleg Mikaylovich Prikhod'ko and engineer Sergey Semonovich Protsenko, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] When comparing programs, the developer usually turns to figures for assistance: block and functional diagrams of algorithms, network and route schedules for completion of work, diagrams of direct components and so on. The need for graph models is natural and is determined by the desire to represent clearly the structure and complex relationships of the program entities.

The traditional display devices for program design (programming languages of different levels) force the programmer to transform preliminary graph models to textual lines. Working with this text, the programmer is forced to periodically restore the reverse conformity of the text-image, which has a negative effect on his labor productivity.

One of the basic principles of effective development of programs, which was developed by Academician V. M. Glushkov in [1, 2]--the principle of clear graphical representation of programs in the form of loaded, oriented P-graphs, is being developed in P-technology. P-graphs were displayed by sequential textual lines up until now in practical programming [3].

The principles of graphical display of programs in P-technology for standard alphanumeric displays and printers (ATsPU) of the YeS EVM [Unified Computer System] and SM-4 computers are developed in the given paper [4, 5]. and effective methods of supporting only graphical display of algorithms in paperless programming P-technology are suggested.

The graphical method of assigning algorithms is based on a specific programming discipline established as a result of experience in working with P-technology. The program graph is a combination of descriptive and executive operators. The descriptive operator defines the structure of the data to be processed or the type of memory of the P-machine (register, counter, table and so on) and is given by an ordinary textual line. The executive operator is a loaded oriented graph constructed from standardized graphic structures.

The four basic graphic structures used in design of executive operators (notations of algorithms) are shown in Figure 1. Each of the structures is a suggested aggregate of characters of the alphanumeric display.

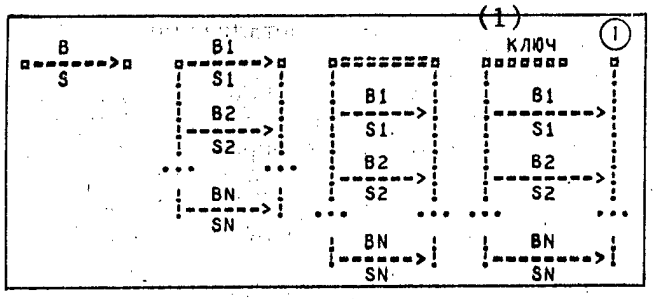


Figure 1.

Key:
1. Key

The structure shown in Figure 1, a is called a P-instruction and is given by the arc above which is written the condition of passage B along the arc (the logical condition) and under which is written the aggregate of actions S (performed if it is a true condition) on data given by the descriptive operators. If condition B is false, the structure in Figure 1, a is considered unfulfillable. In this case an error is recorded in the data to be processed and the transition is made by a program interrupt to the P-program for error neutralization (bypass).

The structure shown in Figure 1, b is called a complex of P-instructions. Completion of the complex includes completion of the single instruction from it in which condition B above the arc is true. The instruction is fulfilled unconditionally by the latter. If this instruction is absent and all the remaining instructions of the complex are not fulfilled, the structure in Figure 1, b is also assumed unfulfillable.

The structure presented in Figure 1, c is called the recurrent complex of P-instructions. The arc without the arrow in it (denoted by the symbol "=") is called the technological arc and corresponds to identification of the vertices joined by it. The structure in Figure 1, c is performed repeatedly until all conditions B on its instructions are false.

The structures shown in Figures 1, a, b, and c are called basic graph structures.

The structure in Figure 1, d is called the key complex of P-instructions and is used to reduce the time of fulfillment of the complex when possible. Fulfillment of the key complex reduces to execution of a single instruction. The ordinal number of this instruction (the instructions are numbered from 1 to n from top to bottom) is determined by the value of the integer function KL, which is calculated prior to fulfillment of the complex. The key complex is

considered unfulfillable if the value of the function is equal to or greater than zero and also if there is a false logic condition of the instruction to be fulfilled.

The executive operators of the graph-program are constructed from the structures considered above by composition of them (concatenation and embedding) similar to how this is done in structured programming. The name by name connected sequence of these structures is called the P-module. An example of a graph-program divided into four P-modules and formulated in the form of a drawing by P-technology [6] is presented in Figure 2.

The main means of creating the graph-program on the display, entry of it, subsequent editing and visual inspection is a graphic dialogue editor. The laboriousness of operations when working with graph-programs is approximately the same as when using traditional text editors. At the same time, the programmer achieves a number of essentially new capabilities due to high structurization of algorithm display on the display screen. The corresponding graph editors in the YeS and SM EVM RTK have their own features that are dependent on the characteristics of the alphanumeric displays of these computers, but the identity of display of graph-programs on both types of machines is retained to the extent possible. The characteristics and capabilities of each of the graph editors are presented below.

A version of an editor for the OS YeS [YeS operating system] supports work with YeS7920 and YeS7906 alphanumeric terminals. The terminal screen is divided into two parts during editing. The visible fragment of the drawing of the graph-program is displayed in the upper part (22 lines for the YeS7920 display and 10 lines for the YeS7966 display). The two bottom lines are used to enter instructions (data) to the editor and to retrieve diagnostic messages.

Two cursors--the physical cursor of the terminal by which instructions are entered and the mathematical cursor (symbol "#") that notes the beginning of the current line of descriptions or the beginning of the current graph structure which is entered, corrected or reviewed--are used when working with the editor.

Any graph structure is entered into the necessary part of the screen in the form of a sequence of instructions corresponding to individual lines of RS/TRAN language operators: ANY, ROW, KEY and so on [3]. The names of the instructions coincide with the names of the indicated key words of this language.

As instructions are entered in the upper field of the screen, the corresponding fragment of the diagram is formulated automatically, individual arcs of the constitutent graphic structures are equalized and the inscriptions on the arcs are also centered. The diagram is shifted with respect to the screen directly by the shift instructions of the mathematical cursor that notes the current graphic structure. Realistically, the mathematical cursor remains in the central part of the screen in this case while the drawing is shifted in a direction opposite to the direction of movement of the mathematical cursor.

The insertion, removal and correction instructions are related to the current graphic structure marked by the mathematical cursor.

The editor provides the following functional capabilities: entry of the diagrams of P-graphs with automatic formulation and equalization of individual arcs of the P-graphs, correction of fragments of P-graphs (insertion, substitution and removal), review of P-graphs on the screen of the terminal by moving the diagram to the left (to the right), upward (downward), retrieval of documentation to the printer in A3-A5 format [6], translation of graph-programs to internal display (RS/TRAN language) and execution of level by level program design with automatic linking of individual levels in the translation and printing phase of graph-programs. The latter function is based on the use of indicators formed by programmers for connections of individual fragments of the program to be designed to the design tree.

All four graphic structures (see Figure 1) and a number of auxiliary structures of RS/TRAN language [3] that facilitate the technology of graph-program design and the continuity of versions of RTK of the YeS OS are realized in the editor.

Work with the graph-programs is carried out by using the following three subsystems in the OS RV [real-time operating system] RTK of the SM EVM:

a graphic dialogue editor that provides entry, visual review and editing of the graph-programs on the basis of the VTA-2000 terminal;

a translator that translates the graph-programs to a form suitable for fulfillment and debugging of programs by traditional RTK devices;

a documenter that permits preparation and retrieval of the graph-program to a printer in some standardized form determined at the phase of technological preparation for production.

The graph structures are edited by using 14 functional keys which are arbitrarily divided into three groups. The first group contains three review keys of the graph-programs (upward, downward and to the right "roller"). The second group includes three editing keys (to remove an element of the graph from the screen, to remove an element of the graph from the memory and end of editing). Finally, the third group of keys combines eight keys for design of graph-programs (beginning of graph, beginning of instruction, beginning of complex, beginning of recurrent complex, next instruction of complex, notation above arc, notation under arc and end of graph-structure).

All symbols of the main keyboard are used in the editor to assign the textual parts of the graph-programs. Unlike traditional editing systems, the functional keys of the first group permit one to review the textual part of the graph-program in the "roller" mode not only in the forward but in the reverse direction as well and to view the graphic part (diagrams) from left to right on the screen.

The functional editing keys (second group) permit erasure of diagrams of the graph-program from right to left from the screen only (the mode preceding

insertion of the graph fragment) or erasure from the screen and from the memory (the mode of removing the graph fragment).

The functional keys of the third group and the keys of the main keyboard are used in design (drawing) of the graph-programs or in insertion of fragments in an already available drawing. Pressing the functional keys results in immediate appearance of the desired fragment of the basic structure on the screen. In this case the cursor automatically moves to the position of predetermination of the next fragment of the structure. The structure is predetermined (writing the logic condition and the operations of action above and below the arc) by using the character keys of the VTA-2000. The data above and below the arc are equalized automatically.

Only the basic graph structures presented in Figure 1 are realized in the editor. The graph-program translator is organized as a syntactically controlled preprocessor to the translator from RTRAN language of the SM EVM.

The main function of the translator--translation of the structures of the graph-programs to RTRAN of the SM RTK--is performed at high speed, while the translator itself is organized recursively and is low in volume.

The graph-programs embedded in the SM RTK comprise the basis of program documentation. The user prepares the text of the graph-programs for printout on the ATsPU in graph form by using a documenter. Preparation includes formatting of the text from standard pages, formulation of title pages, filling in of stamps, review on the display screen prior to printout and making the necessary corrections (see Figure 2).

Let us note in conclusion that programming in P-graphs on a display has a number of advantages compared to ordinary programming using textual lines:

the process of "manual" translation of P-circuits to the texts of P-programs is eliminated and this means there are no additional expenditures on following the conformity between the documentation (P-diagrams) and the texts of P-programs;

perceiving drawings of the graph-programs (rather than textual lines) through a viewing channel, the programmer receives more complete and clearer information about the program and accordingly can more reliably display its operation;

the small dimensions of the screen "provoke" the programmer toward compilation of relatively small drawings of modules, which meets the rules of modular programming and structures the entire program design with higher quality. At the same time, graphic display by efficient use of the screen area permits one to display program modules more informatively and more compactly than in traditional line notation;

there is a new criterion for determination of the complexity of the program in the mnemonic capacity of notations and the clarity and structure of drawings perceived by man "at a glance."

The experience of programming in P-graphs, achieved in the Department of Programming Technology, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, indicates the capability of increasing the labor productivity of programmers in P-technology a minimum of 1.5-fold and reducing the number of errors in program design and the technological effectiveness of using graph-programs as the basic programmer document.

An average of 1 week is required for programmers familiar with P-technology to assimilate the subsystems that support programming in P-graphs.

BIBLIOGRAPHY

1. Glushkov, V. M., "Osnovy bezbumazhnoy informatiki" [Fundamentals of Paperless Informatics], Moscow, Nauka, 1982.

2. Glushkov, V. M. and I. V. Vel'bitskiy, "Programming Technology and Problems of Its Automation," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1976.

3. Vel'bitskiy, I. V., V. N. Khodakovskiy and L. I. Sholmov, "Tekhnologicheskiya kompleks proizvodstva program na mashinakh YeS EVM i BESM-6" [Technological Program Production Complex on YeS and BESM-6 Computers], Moscow, Statistika, 1980.

4. Vel'bitskiy, I. V., "P-Technology-80," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1980.

5. Vel'bitskiy, I. V. and A. V. Moshitskiy, "The Graphic Program Display Interface in P-Technology," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1981.

6. Vel'bitskiy, I. V., "Paperless Programming Technology in a Dialogue Environment," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 6, 1982.

6521
CSO: 1863/72

UDC 519.8

DIOSIM--SYSTEM FOR SIMULATION OF DIGITAL PROCESSES ON YES COMPUTER

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82
(manuscript received 26 Jul 82) pp 55-57

[Article by Candidate of Physicomathematical Sciences Tadeush Pavlovich Mar'yanovich, Candidate of Physicomathematical Sciences Igor' Georgiyevich Kolosovskiy and engineer Nikolay Denisovich Shvab, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] Examples of designing programming systems on the principle of expanding some basic system, specifically, programming systems for simulation of complex systems, are described in [1, 2]. The main advantage of this approach is that there is no need to re-execute a significant part of the work (mainly routine) done in realization of the basic system.

Among investigators that use program simulation equipment in their work, the NEDIS system [3], which has all the devices (properties) of the most developed simulation systems, has become widespread. However, realization of this system only on the BESM-6 computer unjustifiably restricts the number of users. Therefore, the task of developing a simulation system that provides users oriented toward the YeS EVM [Unified Computer System] devices not inferior in their capabilities, ease of use and other indicators to those devices which are provided by the NEDIS simulation system in its digital part, was posed.

The Digital Oriented Simulation System (DIOSIM) was designed as one that meets this requirement. The input language of the DIOSIM includes two groups of devices:

general algorithmic devices of the most developed programming languages (devices for dynamic control of memory distribution, access to data disposition addresses and also devices for working with different types of data and so on);

devices that permit clear and effective description of the simulated entities (devices for description of classes of entities, synchronization of interaction of entities and remote access to data).

In the first case, PL-1 language is used as the basic language and in the second case the corresponding operators of NEDIS and SIMULA-67 languages are used as the superstructure [4].

When combining these two groups of devices in a new language, one should bear in mind that this combination is not a simple increment of the basic language, because the traditional principles of realizing it and the rules of program execution written in this language frequently contradict the requirements of realizing the second group of devices. Thus, for example, activation of an entity, description of the class of which is not external (an embedded class unit), cannot be realized by standard mechanisms for activation of units used in languages with block program structure. A similar situation also arises when the entity control point remains inside the embedded unit as a result of the operator's stopping action.

The incompatibility of the "base" and "superstructure" can be bypassed by the following methods:

modification of the library modules of the basic language interpreter that realize functions in which incompatibility is manifested;

simulation of the properties not provided during realization by external devices of the basic language.

The first method is more cumbersome, but yields more effective results and the second method can feasibly be used in design of the first versions of the system. One can cite realization of the access mechanism to the procedure, the name of which is the variable, as an example of using the first method. The apparatus for working with the based files having an arbitrary number of boundary variables is realized by the second method in the DIOSIM system.

The main devices that distinguish the simulation languages of the SIMULA type were enumerated above. Let us dwell on the main features of realizing these devices in PL-1 language.

The data structure of the class serves as a means to transfer the values of some data from one active status of an entity (class specimen) to another. These structures are naturally represented as based structures by PL-1 language devices. This permits dynamic control of the arrangement of the structure in the main memory and permits one to find the address reference to its location.

The structure is arranged at the moment of passage through the entity and is freed if there is no reference to this entity. From the programmer's viewpoint, these operations should be performed by the systems method, i.e., the same as for automatic data localized within the procedure unit in PL-1 language. The so-called disconnect point is created for this purpose within the body of the class, which is an analog of the procedure body. Part of the class body up to the disconnect point is executed in generation of the entity. The data structure of the entity is formatted and arranged in this part. In all the remaining cases, processing of the class body upon activation of the entity begins after the disconnect point. The structure of the given entity is actuated upon each entry into the body by recording the address of its location and also of the structure of the enveloping entities.

The sequence of events occurring in the simulated system is controlled by means of a schedule, which is a list structure. As in any other list, the entities are represented by their own certificates in the schedule. The certificate is a based data structure among whose elements there are address references to the certificates of the receiving and predecessor entities (if these exist) and also to the data structure of the entity. The certificate is a device that permits the same entity to be placed in different lists simultaneously. References to the certificates are made by variables of the same type, which are a special case of indicator variables of PL-1 language.

PL-1 language permits one to denote different data by coincident identifiers if they are elements of different structures. To distinguish these data, they are refined by the names of their oldest structures. Several structures with the same name, differing only by their location, can be present simultaneously in the model. These are, for example, data structures of different entities of the same class.

Refinement by the name of the oldest structure (by the name of the class) is inadequate to distinguish elements of these structures of the same name. Name refinement, which refers to a specific specimen of the structure by means of a name expression, serves this purpose in SIMULA type languages. Some name A is refined by means of PL-1 language using fixation of the location address of the oldest structure for element A (by indirect reference by the name qualified variable) with subsequent refinement of name A by the name of this oldest data structure.

For example, let there be described class C and the name variable N, qualified by this class, in the model

```
C:#CLASS;
DECLARE A;
.
.
.
END C;
.
.
.
DECLARE N#NAME(C);
```

Then the construct

```
...N .. A...
```

is represented in the following manner by means of PL-1 language:

```
...N C.A...
```

Joining some class C to an entity can be simulated by means of PL-1 language by organization of a unit inside which is placed the only description--the

data structure of class C with subsequent fixation of the location address of this structure. In this case there is no need to refine the names used within the connection unit since each of them is either determined clearly as an element of the connected data structure or is described outside the connection unit. A chain of name refinements of arbitrary length and the arbitrary depth of embedding the connecting units can be realized by these devices.

It was stated above that the data structure of the class is represented in PL-1 language as a based structure. According to PL-1 syntax, a based structure cannot contain based and certain other types of data as its own elements. The elements of the class data structures of these types are represented in based structure by indicator type data in PL-1 language that provide the location addresses of the corresponding data or the procedure and class input addresses that are elements of the data structures.

DIOSIM input language is oriented toward high-level PL-1 language. It includes the entire PL-1 language as its own subset. Orientation toward PL-1 language denotes the coincidence of formats of the corresponding PL-1 structures and DIOSIM language. New key words begin with the symbol "#," which is not recommended for use for other purposes. The class and connection units, the titles of which are written as follows, are added upon expansion of the basic language:

```
class title::=name-class:#CLASS (list of formal parameters);

connection heading::=#INSPECT expression-name-qualified DO;
```

The class body is described the same as the procedure body, while the connection unit body is described as the body of the BEGIN unit. Embedding of the units is permissible the same as in PL-1 language. The visibility of barriers is the same as in all languages with unit structure, but the names of data that are elements of the class data structure, to the specimen of which connection is made, and then the names of data described in the connection unit and described by generally accepted rules in the enveloping units are primarily visible within the connection unit.

The same as in PL-1 language, descriptions can be located in any point of the unit. However, in the case of a class unit, only the data, descriptions of which are located between the heading of the class and the first encountered operator, are included in the data structure of this class. The remaining data become accessible only as a result of natural activation of the given unit (i.e., upon access to it through the heading or one of the inputs).

If the different data described by identical identifiers are elements of the data structures of different classes, then they can be distinguished by means of name refinements

```
name-refined::=expression-name-qualified..identifier
```

The name qualified expression should refer to the entity to which the refined name belongs. It can either be a name qualified variable or a function that

returns the reference to the entity or a structure that describes generation of an entity

#NEW name-class

Descriptions of the name variables and histograms are introduced along with the descriptions of the class units to supplement the descriptions of PL-1 language in DIOSIM language

description-variable-name::=name-variable#NAME [(qualification)]

If the name variable is qualified, then the name of the class emerges as the qualification. The qualification is used during translation to check the legality of the refinement and to organize access to the data structure of the qualifying class

description-histograms::=name-histograms (left-boundary space right-boundary)

When working with histograms, the tabulation and histogram printing operators are used

operator-tabulation::=#TABULATE expression-real#IN name-histograms;

operator-print-histogram::=#PRINTH name of histogram;

The interaction of entities is synchronized by using pause, activation and passivation operators

operator-pause::=#HOLD expression-real;

operator-activation::=#ACTIVATE expression-name [{#AFTER #BEFORE} expression-name #DELAY expression-real];

The synchronization operators bring order to the activation sequence of entities by priority and time, placing them in the schedule, which is a cyclical list. These lists can be organized in the system by using the #LIST procedure. If A is described as a name variable, then the name confer

A = = # LIST;

causes organization of the list to the vertex of which A refers.

The following operators are available for working with lists

#INSERT expression-name {#AFTER #BEFORE} expression-name;

#DELETE expression-name;

which accordingly enter or remove from the list the element to which the name expression refers. If A is an element of the list, then #SUCH (A) indicates

the element of the list following A, while #PRED(A) indicates the element that precedes A.

There is a generator of a sequence of pseudorandom numbers equally distributed in the range (0, 1) and a set of procedures that convert this distribution to some standard numbers (uniform in the range (A, B), exponential, normal and so on) in DIOSIM.

The routing system permits one to follow the development of processes in the model during debugging of it. In this case the following events are recorded: activation, pause, passivation and completion of an entity and also the status of the active and generated entity. Routing can be complete or partial. In complete routing, all the enumerated situations are registered and with partial routing, one can select some subset of situations and also the names of the classes to be routed and the indicators to the entities to be routed. Routing can be carried out by the operator #TRACE at any point of the program and can be cancelled by the operator #NOTRACE.

Errors in translation and in execution of the model program are diagnosed by the corresponding messages.

All the DIOSIM programs are stored in text and load module libraries. Functionally, this consists of three parts: translator and interpreter libraries and also task libraries. The latter contains the set of tasks (or of catalogued procedures) that provide editing and translation of texts, formation of load modules, accounting (simulation), copying and compression of libraries.

BIBLIOGRAPHY

1. Gorbatenko, D. D. and A. K. Polyakov, "Yazyk modelirovaniya slozhnykh protsessov, baziruyushchiysya na PL-1" [Language for Simulation of Complex Processes Based on PL-1], Moscow, SNPO Algoritm, 1980.

2. Manovitskiy, V. I. and Ye. M. Surkov, "Sistema imitatsionnogo modelirovaniya diskretnykh protsessov (DISM)" [A Digital Process Simulation System (DISM)], Kiev, Odessa, Bishcha shkola, 1981.

3. Glushkov, V. M., V. V. Gusev, T. P. Mar'yanovich and M. A. Sakhnyuk, "Programmnyye sredstva modelirovaniya nepreryvno-diskretnykh sistem" [Programs for Simulation of Analog-Digital Systems], Kiev, Naukova dumka, 1975.

4. Dahl, W. I., B. Murhaug and K. Newgord, "SIMULA-67: universal'nyy yazyk programmirovaniya" [SIMULA-67: A Universal Programming Language], Moscow, Mir, 1969.

6521
CSO: 1863/72

UDC 658.012.011.56:631.3.069

DISPLAY-ORIENTED EDITOR OF SYMBOLIC DATA FOR SM-3 AND SM-4 COMPUTERS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 manuscript received 8 Dec 81, after revision 30 Jun 82) pp 66-67

[Article by Tat'yana Valerianovna Lunina, junior scientific associate, Institute of Automatics, Kiev]

[Text] Existing standard editors of symbolic data for the SM-3 and SM-4 computers offer a wide range of capabilities to the user for text correction. However, the nature of editing itself is artificial since it reduces to manipulations with specific formal symbols which are rather strict and cumbersome. The latter is explained by the considerable number of instructions entered to expand the capabilities and flexibility of the editor. The need to use these instructions requires definite precision and accuracy from the user, especially when working with long lines and files.

Moreover, the affiliation of the given editors to the class of coordinate type editors makes precise indication of the location of the correction in the form of the coordinate in one or another system necessary.

Thus, for example, by using the standard EDIT editor to replace a group of symbols, one must move the cursor to the editing position (for which two shift operators must be given and the operator counts the number of lines and number of symbols that figure in the given operators) and one must indicate in the substitution operator the new text and the number of symbols to be replaced. Moreover, data must be periodically called to the display screen and the SEND key must be pressed at the end of each operator.

Because of the foregoing, inconveniences arise that reduce the labor productivity of the programmer and that result in inefficient expenditures of machine time.

The DIOR (Display-Oriented Editor) symbolic data editor for the SM-3 and SM-4 computers, that provides direct correction of text on the display screen by using hardware-realized editing devices, is presented in the given article. It should be noted that this method of editing corresponds to the greatest extent to the established human skills of working with pencil and paper and is therefore very convenient to use [1-3].

The developed operational editing program is oriented toward the standard VTA-2000 or Videoton alphanumeric display, takes into account the functional modes of the given display and its extensive capabilities of text editing on the screen, is very simple to operate and is devoid of the deficiencies noted above for standard editors. Moreover, a number of characteristics of the given editor that reveal its advantages compared to a standard editor should be noted:

the file to be edited (the input file) is simultaneously the output file. There is no need in this case for the user to copy segments of text not subject to changes from the input to the output file. The result is greater "transparency" of the editor to the user simultaneously with reduction of the required set of operators;

the minimum possible set of operators that include input-output operators to peripheral devices and the page "listing" operator on the display screen;

the maximum simplicity of display control both during transmission of edited data to the computer memory and during transition to the autonomous editing mode.

The input and output data for the editor can be placed both on magnetic disks and on papertape. Moreover, there is the capability of entering the text file directly from the screen of the alphanumeric display.

The essence of editing includes sequential retrieval of a specific part of the text file to the display screen, the so-called data frame, entry of the required changes in the text displayed on the screen by using the functional display keyboard and entry of the corrected frame in the initial file. The data frame is understood here as 20 sequential lines of the text file.

The given editor has the following functional capabilities:

entry of the file to be edited in the main memory from papertape or from magnetic disk (operator R);

entry of the text from the screen of the alphanumeric display;

review of the file to be edited by frames: from the beginning of the file, from the operator whose number is indicated in the directive, beginning with the line that includes the text given in the directive (operator P);

callup of the next frame of textual information to the display screen (operator L);

correction of the current frame initiated on the screen and storage of it in the initial file;

output of the edited file to magnetic disks or to papertape (operator W).

Prior to operation of the program, one should call the monitor of the operating system, enter the necessary date and start the editor with the standard monitor instruction

R_DIOR,

where DIOR is the name of the editor.

Retrieval of the message RECEIVE to the display screen, which will subsequently always be retrieved after processing of the control directive given by the user, indicates the capability to begin work. After starting the editor, the user should enter the initial file in the main memory from papertape or magnetic disk, using one of the following directives:

R, PR:******.MAC
R, DK:******.MAC

where PR and DK are the logic names of the papertape and magnetic disk input device, respectively, and ******MAC is the name of the file and expansion of it.

If there is no text file in any of the carriers, the text can be entered directly from the display screen. To do this, one should convert the display to the autonomous mode by pressing the FREE keys and one should turn to the set of necessary data. The lines are separated from each other by the symbol PS. The selected text is then entered in the computer memory by using the keyboard.

After the initial file is in the main memory, it becomes accessible for editing. To do this, the editor retrieves the first frame of the file to be corrected to the display screen by using the control directive

P[,N] or P[,text]

where N is the number of the operator in the initial file and text is an arbitrary byte combination.

As a result, 20 sequential lines of text appear on the screen. The user can then enter the necessary changes in the text displayed on the screen by means of the keyboard after converting the display to the autonomous mode. Thus, to substitute a group of symbols (see the example given above) the keys for moving the cursor must be pressed once or twice and the keys for shifting the text and the set of necessary symbols must be pressed once. In this case there is obviously no need for a second callup of data to the screen to check it.

The capability of inserting supplementary lines is provided by the excess number of lines of the screen with respect to the retrieved frame. Moreover, one can make use of several operators on the same line by using the symbol VK as the separator.

Upon completion of editing the frame, the user dispatches the corrected data from the buffer memory of the device to the computer by sequential pressing of the FREE, DUPLEX and TRANSMISSION keys. As a result, the edited text will be inserted into the initial file instead of the old text and 20 sequential lines will be displayed on the display screen from the beginning of the frame to check the entered changes.

If the given frame requires no further correction, the next data frame is retrieved by using directive L and the editing process is repeated. When the entire initial file is corrected in the necessary manner, it is entered on papertape or on magnetic disks by one of the following operators:

```
W,PR:******.MAC
W,DK:******.MAC
```

If it is unnecessary to make any manipulations with the given file, editing is completed by the directive E.

The DIOR editor now functions in the single-console mode under the control of the FOBOS (RAFOS) operating system on the SM-3 and SM-4 computers, which determine the selection of the mnemonics of the control operators. The required main memory capacity comprises from 12 to 28 Kbyte. Work is being conducted simultaneously on modification of editors for the OS RV [Real-time operating system], which permits it to be used in the multiconsole mode.

Operation of the DIOR editor at the Institute of Automation (Kiev) confirms its high efficiency, convenience of use and simplicity of mastery.

BIBLIOGRAPHY

1. Zaytsev, V. G. and V. T. Yegizar'yan, "Operational Editing of Programs with Automated Design of ASUTP [Automated Production Process Control System] Programs Editing," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 3, 1981.

2. Johnson, K. and T. Costlow, "Automated Development of Applied Programs," ELEKTRONIKA, Vol 54, No 7, 1981.

3. Johnson, K., "Expanding the Use of the UNIKS Operating System," Loc. cit.

6521
CSO: 1863/72

UDC 681.3.06:519.8

RELATIONAL HOMOGENEOUSLY-STRUCTURED DISCONNECTED DATABASE

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82
(manuscript received 17 Mar 82, after revision 30 Aug 82) pp 74-78

[Article by Doctor of Technical Sciences Nikolay Georgiyevich Zaytsev, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] Principle features of the suggested database (BD). A pronouncement is made as an elementary logic unit of data that expresses the relationship between two entities. A formally-structured representation of the pronouncement in the form of data was considered in detail in [1] and is called the display element (OE). The simplest form of grouping is a set of display elements containing all pairs of terms of two classes, between which a given ratio is observed. This set of pairs also emerges as a definition of ratio in set theory [2].

Since description of the subject area is the aggregate of all pronouncements about it, combined (or not combined) into groups, the set of ratios is a possible and sufficient form of describing the subject area in a database.*

All ratios, with the exception of symmetrical ratios (for example, "differences"), are directional in nature; therefore, along with direct ratios, one should also consider inverse ratios (for example, "less" and "more"). There are no bases whatever to give preference to either a direct or inverse ratio. Thus, "coworkers" are of primary significance for personnel policy in the ratio of "speciality of coworkers," while "specialties" are of primary significance for production activity. The name of the inverted files widespread in practice in databases itself emphasizes the secondary nature of these files (if one does not talk about secondary digit capacity) and at the same time emphasizes their non-compulsory nature.

It is feasible to include both direct and inverse ratios in databases on identical rules, which guarantees at least three advantages to this database:

* We feel that ratios are the only possible form of descriptions in a database; therefore, to talk about its relational nature makes no sense. Nevertheless, since this viewpoint is not yet generally accepted, we will talk about a relational database.

equality in representation of all classes of entities of the subject area;

logic completeness of representation of all ratios observed in the subject area in the database;

symmetry of the database for all possible applications.

If the first two advantages are important from the conceptual (logic) viewpoint, the third advantage is of primary significance for the applied (technological) effectiveness of the database.

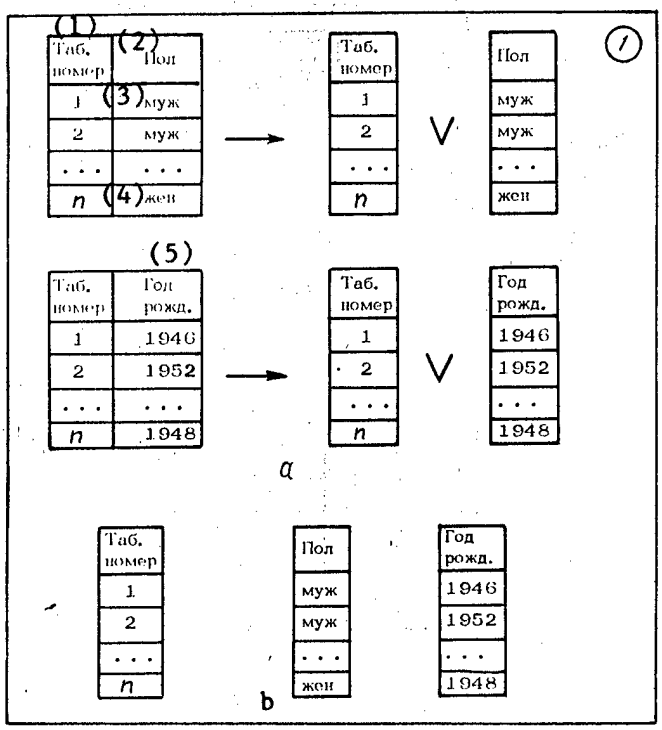
One can take exception to the fact that so-called redundancy will occur in this database, but it should be understood in this case what the nature of this redundancy is. It is known that development and use of databases require the expenditure of three types of resources--human labor, machine time and memory carriers. Without going into detailed consideration, one can state boldly that human labor expenditures (primarily on recruitment and training) and machine time expenditures (mainly due to search and retrieval) are reduced when using the proposed database. The capacity of the memory increases in this case, but the cost in modern data processing systems comprises such a negligible fraction compared to the total cost of these systems that it simply makes no sense to talk about this.

It is usually accepted to connect the terms of the ratio in a database into classes, the constituent components of which are "recording" (lines of tables or processions) [3]. The reasons (and very convincing ones) for this linking are first, all human experience in working with documents consisting of lines, second, the need for an entire group of ratios in solving an applied problem, third, the convenience of operating with a single class instead of several and fourth, the specific economy of memory due to a brief general instruction for all ratios of the first entity.

Nevertheless, despite the traditions of linking, let us take the opposite tack and "dissect" the ratio into two semiratios, in the first of which will be the first terms of the pairs and in the second of which will be the second terms. As a result of this "dissection," we will have two disconnected semi- ratios, the presence of a ratio between which must be registered separately, for example, in metadata (Figure 1, a). Having carried out a similar proce- dure with all ratios, we find a set of semiratios. Some of them are identi- cal and they can not be repeated in the database (Figure 1, b).

Modern programming equipment provides simultaneous joint processing of many classes of data [4]. The features of access in the computer memory do not require adjacent location of terms of the same pair. These two capabilities permit one to "link" two semiratios of the same ratio into a single whole dynamically--at the moment of processing. Joint consideration is provided by the factor of simultaneous reading itself of the necessary semiratios into the main memory.

The reasons for preliminary static linking are no longer valid. First, the tradition of the "neighborhood" of terms of the ratio is determined by the



"Dissection" of Ratios Into Semi-Ratios (a) and Storage of the Latter in the Computer Memory (b)

Key:
1. Table number
2. Sex
3. Male
4. Female
5. Year of birth

features of human vision (continuity of eye movement), whereas access to an adjacent and to a remote cell is identical in a computer. Second, the group of ratios required to solve an applied problem is guaranteed, but is guaranteed dynamically. Third, the number of classes being processed simultaneously in automatic control is of no significance on the basis of metadata. Finally, fourth, the semiratios common for all ratios are registered once in the database.

At the same time, a disconnected database achieves decisive advantages compared to a traditional database consisting of recordings:

this database consists of the simplest homogeneous classes--sequences of identical elements;

it is identically effective to all applications, regardless of the required combination of ratios.

One can say in a specific sense that the proposed Relational Homogeneously-Structured Disconnected Database (ROSS BD) is the limit in achieving logic understanding and structural elementariness.

Composition of the ROSS BD. Based on study of the subject area, all those classes of entities which will be described in the database are determined and registered. Let us emphasize that both material and abstract entities--that which is called properties--are enumerated in the composition of entities. The results of enumeration are represented in the display element format in the following manner:

⟨Subject area⟩ ⟨includes⟩ ⟨class$_1$ of entities⟩;
⟨Subject area⟩ ⟨includes⟩ ⟨class$_2$ of entities⟩;
...
⟨Subject area⟩ ⟨includes⟩ ⟨class$_n$ of entities⟩.
⟨Class$_1$ of entities⟩ ⟨includes⟩ ⟨entity$_{11}$⟩;
⟨Class$_1$ of entities⟩ ⟨includes⟩ ⟨entity$_{12}$⟩;
...
⟨Class$_1$ of entities⟩ ⟨includes⟩ ⟨entity$_{1m}$⟩.
...
⟨Class$_n$ of entities⟩ ⟨includes⟩ ⟨entity$_{n1}$⟩;
⟨Class$_n$ of entities⟩ ⟨includes⟩ ⟨entity$_{n2}$⟩;
...
⟨Class$_n$ of entities⟩ ⟨includes⟩ ⟨entity$_{np}$⟩.

Thus, in the mentioned example (see Figure 1), the subject area includes the following classes of entities: the table number, sex and year of birth.*

The composition of the entities of each class can be given both by simple enumeration (sex--male or female) and by indication of features of affiliation to a given class (table number--from 00001 to 99999).

The results of accounting (inventory) of entities of the subject area reflected in the data comprise the first part of the database--let us call it the entity part of the database. It performs the following functions:

it determines the initial classes for the ratios that reflect the subject area;

it contains the composition of these classes.

Structurally, the entity part of the database contains the classes of entities in which the entities contained in these classes are enumerated. The entities in the classes are usually ordered and each class can be divided into subclasses (Figure 2).

The next aspect of consideration of the subject area are the ratios between the different pairs of entities observed in it. We feel that one should proceed from the following statements in this case:

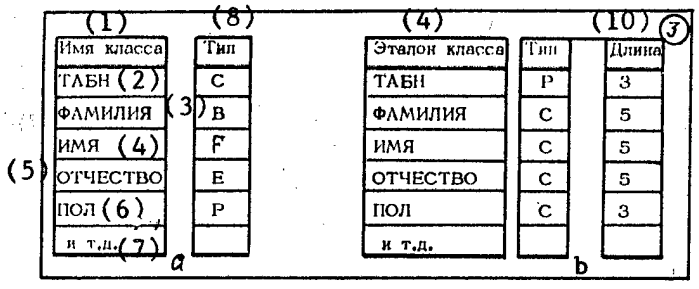
* It would be more correct to name each class in a multiple number, but in documents they are traditionally indicated in a single number.

It is obvious that the class that generates the ratios can be indicated once (see Figure 1, b).

Thus, the second part of the database reflects the ratios existing in the subject area. Let us call this part the relational part.

The resulting database itself is the subject area and also needs description, which is given in metadata [4]. The database entities are considered first, i.e., the classes of entities and all semiratios (one must turn attention here to the single significant feature of the description: although the classes are enumerated, only one term of each class--its reference--is actual- ly described due to homogeneity). Abstract entities that characterize the properties of the data (for example, dimension, type and so on) are also enumerated along with this. We note that these entities of the subject area "data" comprise the basis of the data description language [5].

Thus, the third part of the database are metadata about the entities--the metaentity part of the database (Figure 3, a).



Examples of Metaentity (a) and Metarelational (b) Data

Key:

1. Name of class
2. Table number
3. Surname
4. First name
5. Patronymic
6. Sex
7. And so on
8. Type
9. Class reference
10. Length

The ratios between classes of data and their properties are presented in the fourth part of the database--let us call it the metarelational part of the database (Figure 3, b). Strictly speaking, description of the metaentity and metarelational parts also requires description, but the corresponding descriptions materialize directly in the programs for working with metadata since the structure of the metadata is clearly determined.

Finally, all the relationship between the semiratios of the subject area must be registered, which comprises the fifth part of the database which we shall call the connected part of the database (Figure 4). The structure of this part corresponds to that of the relational parts: there is a class of ratios that are called the same as in the subject area (for example, ⟨employee⟩ ⟨has⟩ ⟨wage⟩). It is indicated in the description what is included as the first

Example of Enumeration of Entities in Classes

Key:

1. Sex
2. Male
3. Female
4. Nationality
5. Russian
6. Ukrainian
7. And so on
8. Specialty
9. Programmer
10. Operator
11. Position
12. Technician
13. Engineer
14. Party Affiliation
15. Member of CPSU
16. Komsomol member
17. No affiliation
18. Education
19. Higher
20. Secondary
21. Primary
22. Family situation
23. Married
24. Single

all ratios reduce to binary ratios;

the set of all possible ratios is rather limited in the real world.

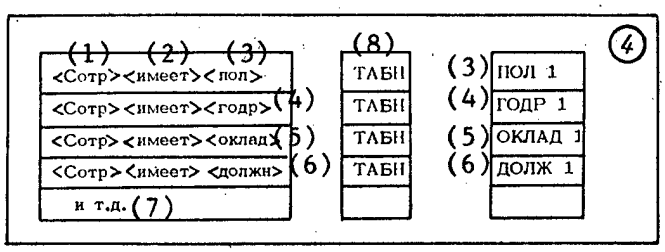
Using the composition of entities of the subject area, divided into classes, one can employ the following constructional algorithm for finding a set of ratios. Each class is considered alternately and all other classes are sequentially compared to it. If there is some ratio between the terms of the pair of classes to be considered, then it is registered. It is important in this case that each class, regardless of the fact whether it participates in the previous ratios or not, be considered as the one that produces the ratios.

The following set of ratios is found as a result (each line indicates the ratios which includes the class to be considered as the first term and in this case several ratios may exist for the same pair of classes):

$$K_1R_1K_2,\ K_1R_2K_3,\ \ldots,\ K_1R_lK_j,\ \ldots,\ K_1R_rK_n;$$
$$K_2R_2K_1,\ \ldots,\ K_2R_jK_m,\ \ldots,\ K_2R_rK_n$$
$$\ldots$$
$$K_nR_1K_1,\ \ldots,\ K_nR_jK_{n-1}.$$

semiratio and what is included as the second semiratio in this ratio and inverse ratios are also presented. Let us turn attention to the fact that the data comprising the semiratios should have a name different from the enumerating class (they are denoted by ones in Figure 4--sex 1, year of birth 1 and so on).

Thus, the database consists of five parts, each of which consists in turn of the simplest classes--of sequences of elements of the same type.



Connected Data

Key:

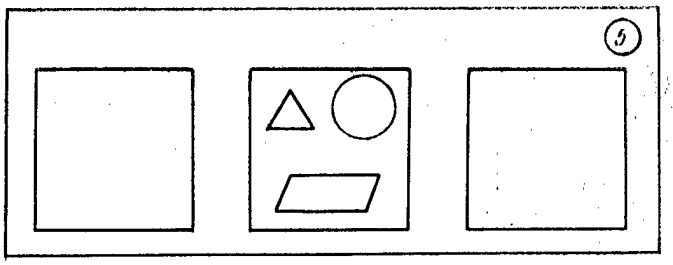
1. Employee
2. Has
3. Sex
4. Year of birth
5. Salary
6. Position
7. And so on
8. Table number

Representation of ratios. In the real world each of the entities is found in many ratios and in this case with many entities in each ratio. In the sphere of language and logic, both entities and ratios are expressed in names (data), which permits one to reproduce the same entity or the same ratio repeatedly essentially without expenditures. As a result, the complex relationship of the entity to the medium can be noted by a set of unit ratios with unit (but not necessarily with atomic) entities. The entity to be considered is reproduced according to the number of all unit ratios, while each ratio is reproduced according to the number of entities with which the entity to be considered is in a given ratio. These entries are also judgments.

The so-called geometric subject area is presented as a clear example in Figure 5. The ratios in which "square 2" is located are noted in the following manner:

⟨square 2⟩ ⟨is equal to⟩ ⟨square 1⟩;
⟨square 2⟩ ⟨is equal to⟩ ⟨square 3⟩;
⟨square 2⟩ ⟨includes⟩ ⟨triangle⟩;
⟨square 2⟩ ⟨includes⟩ ⟨circle⟩;
⟨square 2⟩ ⟨includes⟩ ⟨parallelogram⟩.

Unlike reality where the ratios are bound to the entity (not physically separated from it), the identical unit ratios in the database are combined into groups. Entities can be represented by the reproduced names in different ratios (see Figure 1, a).



Geometric Subject Area

The grouping of unit ratios, which are identical, in the database determines the formation of the first semiratios--they are the classes and enumerating entities to which the given ratio is inherent. Thus, the class of "employees" is the first semiratio for the ratio "has" (see Figure 1, a). Three cases are possible in this case:

1) Each of the abstract entities (properties) is in a given ratio only to one entity and semiratios (terms of each pair of ratios) are linked by simple equality of ordinal numbers (this is expressed in Figure 1, b by the location of those semiratios on the same line, i.e., by a common ordinal number).

2) Each of the entities is in a given ratio with several entities, which is typical for ratios between real entities (for example, for inclusion ratios). One can repeat this entity by the number of entities and the entity class is no longer suitable for the first semiratio. The auxiliary semiratio containtaining numbers equal to a corresponding number of terms in the second semiratio can be placed in conformity to this class.

3) The limiting case occurs when an entire class of entities corresponds to a single entity (for example, when solving problems related to classification). In this case the first term is simply indicated in the description of the ratio in the metadata.

One should bear in mind the three essential circumstances when describing ratios.

First, although it is formulated from a specific entity class, all the terms of this class do not have to be included in the second semiratio, all the terms do not have to be included one time each and all terms do not have to be included in ordered form.

Second, elementary data do not have to emerge as its components in semiratios. Thus, for example, the day, month and year is indicated in the constructive entity "date" and the number and type of specific parts is indicated in the "inclusion" ratio.

Third, only direct ratios are given in the database, i.e., ratios occurring directly between two entities to be considered. Intermediate ratios (through an intermediate entity) are not registered since there can be very many of

these ratios (there are always considerably more combinations than initial elements) and they can be determined on the basis of direct ratios.

Formation and restoration of the ROSS BD. A natural condition for functioning of the ROSS BD is working with documents that meet the requirements of machine processing systems. Data are converted from input documents essentially by the same algorithms as in [6], with the exception of the last phase: the data are distributed in the form of semiratios rather than in the traditional form of notation by lines. One can also postulate more simply, having formed the entries by the traditional technique, they are then converted to semiratios, i.e., the sequence of entries in a sequence of elements is simply "dissected." Let us emphasize that there is a single sequence in the traditional form of line by line entry, while several sequences of elementary data are produced in the ROSS BD, which is also the main feature of formation of the ROSS BD.

Another significant feature of the ROSS BD is parallel formation along with direct and inverse semiratios, which was already discussed.

Main procedures of the ROSS BD. The following procedures were determined for working with ROSS BD on the basis of preliminary analysis of their structure and the practical experience of developing data processing programs [4, 6]: shifting of classes of data, orientation in disposition of data, enumeration of elements of a class, search, development of intermediate ratios and theoretically multiple operations.

The proceudres of shifting of data classes are writing and reading operations from external stores. The natural requirements on this procedure are joint work with many classes and work with classes divided into subclasses.

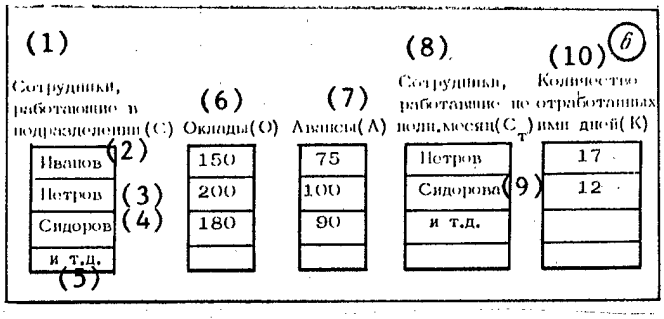
Since the elements of semiratios, considered jointly in the main memory, are arbitrarily located, it becomes necessary to determine their addresses, which is also accomplished by the procedure orientation in disposition of data, which determines the address of the element by the name of the class and the ordinal number of the element in it.

The main procedure when working with data classes is enumeration of the elements of the class. According to description of the class structure, which can be further divided into subclasses, all elements of the given class are issued sequentially for processing. This procedure should be a re-enterable procedure (the capability of moving out of and into the procedure at any phase of enumeration should be provided), since the terms of many classes are enumerated simultaneously. The inverse procedure is the procedure of creating the class when the elements of one or several classes to be processed are combined into a new class.

The search procedure is necessary upon comparison of classes when creating intermediate ratios. Both sequential review and dichotomous search must be provided in the algorithm of this procedure.

As already indicated, only direct ratios are stored in the ROSS BD--intermediate ratios are created by using the corresponding procedure. To do this, all

the semiratios of the initial ratios are entered in the main memory and a "chain" that determines the ratio between the first and last semiratios is established between them by the mentioned procedure. Theoretically multiple operations--both single- and two-place--are necessary upon comparisons [1].



Entity Data in ROSS BD for Calculation of Salary

Key:

1. Employees working in subdivision (S)
2. Ivanov
3. Petrov
4. Sidorov
5. And so on
6. Salaries
7. Advances
8. Employees not working a full month (S_t)
9. Sidorova
10. Number of days worked by them (K)

Task programming. The input language of the ROSS BD must be created to program applied problems, which may be similar in general features to the input language of the OSMO [General systems software] [4], which contains procedures on classes and elements of data. With regard to the variation of the principles of database organization, the composition of the operators on data classes in the input language of the ROSS BD should be different, although these operators will be based on the same theoretically multiple concepts as in the input language of the OSMO. The element operators essentially coincide in both languages.

The problem program consists of class processing formulas and element processing formulas. Upon translation, the class processing formulas are converted to ratios to class processing procedures, while the element processing formulas are converted to the corresponding sequences of macrooperators. Translation is carried out on the basis of the data stored in the metadata of the ROSS BD.

Two features must be taken into account when developing a program in the input language of the ROSS BD:

1. Since the elements in the element processing formula are not connected in the entry, a program mechanism must be provided which addresses to the necessary semiratios according to the names of the elements in the formula.

2. Since there can be a different number of terms in different ratios belonging to entities of the same class, the nonequivalent semiratios must be "equalized" so that the elements related to the same entity in different semiratios are related to the same entity (for example, only those employees who worked an incomplete number of days in the month rather than all employees can be included in the wage calculation table).

As an example, let us consider the simplest case of wage calculation in proportion to the number of days worked with deduction of the income tax and distributed advances. The input data are the ratios presented in Figure 6. The income tax is calculated by using procedure P.

The problem is formulated in the given case as follows: issue a list of wages calculated by the formulas:

a) $z = o-P(z)-a,$

if the employee worked a full calendar month;

b) $z=o\frac{k}{M}-P(z)-a,$

if the employee worked an incomplete calendar month (M is the number of working days in the month).

The program is written as follows in the input language of the ROSS BD:

LIST: S, O, A, S_t, K.
CALCULATE: if $s = s_t$

$z=o\frac{k}{M}-P(z)-a,$

otherwise

$z=o-P(z)-a.$
PRINT: S, O, K, A, Z.

BIBLIOGRAPHY

1. Zaytsev, N. G., "Printsipy informatsionnogo obespecheniya v sistemakh pererabotki informatsii i upravleniya" [Principles of Information Support in Data Processing and Control Systems], Kiev, Naukova dumka, 1976.

2. Kemeny, J., J. Snell and J. Tompson, "Vvedeniye v konechnuyu matematiku" [Introduction to Finite Mathematics], Moscow, Izdatel'stvo Inostrannoy literatury, 1963.

3. Deyt, K., "Vvedeniye v sistemy baz dannykh" [Introduction to Database Systems], Moscow, Nauka, 1980.

4. Zaytsev, N. G., "Obshchesistemnoye matematicheskoye obespecheniye EVM tret'yego pokoleniya dlya obrabotki dannykh" [General Systems Software of Third-Generation Computers for Data Processing], Moscow, Statistika, 1980.

5. "Yazyk opisaniya dannykh KODASIL" [Language for Description of CODACIL Data], Moscow, Statistika, 1981.

6. Zaytsev, N. G., "Informatsionnoye i matematicheskoye obespecheniye ASUP" [Information Support and Software of Automated Enterprise Management Systems], Kiev, Tekhnika, 1972.

6521
CSO: 1863/72

SCIENTIFIC AND TECHNICAL COMMITTEE ON COMPUTER PROGRAMMING LANGUAGES AND SYSTEMS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 111-112

[Article by engineer Aleksandr Nikolayevich Maslov, USSR State Committee for Science and Technology, Moscow]

[Text] The Scientific and Technical Committee on Computer Programming Languages and Systems was created in April 1982 by decree of the USSR State Committee for Science and Technology. The chairman of the committee is corresponding member of the USSR Academy of Sciences A. P. Yereshov. The main tasks of the committee are:

preparation of proposals on problems of development, unification, standardization, realization, introduction and accompaniment of programming languages and systems;

participation in formulation of work schedules (scientific and technical, publishing and personnel training) and consideration of technical assignments for development of programming languages and systems;

expert analysis of documents on programming languages and systems.

Organization of working groups with recruitment of specialists that are not members of the committee has been organized to fulfill these tasks of the committee. The following working groups are already operating or being organized:

1. For ALGOL-68 programming language (for introduction and coordination of developments).

2. For ADA programming language (for coordination of developments and determination of the need for the language).

3. For specifications langauges (especially with regard to the problem of total loading of supercomputers).

4. For programming languages for mini- and microcomputers.

5. For working out requirements on computer programming systems.

6. For standardization of programming languages.

The presently standardized languages are ALGAMS (GOST 21551-76), FORTRAN (GOST 23056-78), basic FORTRAN (GOST 23057-78) and COBOL (GOST 22558-77). The working group for programming languages for mini- and microcomputers is considering proposals on standardization of PASCAL, COBOL-M (for minicomputers), BASIC and other languages.

According to data of a selected questionnaire survey of 1970, the most popular language in the USSR was ALGOL-60. According to data of the same survey of 1980, the use of languages is characterized by the following data: PL-1--35 percent, Assembler--20 percent, FORTRAN--17 percent, COBOL--15 percent, other languages (including ALGOL-60)--13 percent. The use of BASIC comprises no more than 2 percent. However, the situation is beginning to change with regard to production of new types of computers (especially minicomputers) and the appearance of new areas of computer application; therefore, a need has arisen to gather and consider proposals on standardization of new languages.

The data of the questionnaire survey and the analysis of specialists indicate the need to upgrade requirements on programming systems, which is also the task of the working group on working out requirements on computer programming systems. Organizations and specialists desiring to participate in the activity of this working group can send their proposals in a form suitable for xeroxing (preferably double-spaced). The following can be outlined in the proposals: general requirements on the programming system (as the combination of programming devices that provide development and fulfillment of programs on the computer according to some technology), requirements on individual components of programming systems, individual principles of developing programming systems (for example, ergonomic), generalization of Soviet or foreign experience and surveys of the literature, correlation with other problems (for example, with design of input-output devices) and other problems.

The proposals should be sent to the address: 103109, Moscow, GSP, ultisa Gor'kogo, 11, GKNT, GUVT i SU, Tsoy Viktor Nikolayevich.

Other contacts with the committee should also be made through the same address.

6521
CSO: 1863/72

UDC 619.682.5

KIEV AUTOMATED CONTROL SYSTEM, DESIGN FEATURES AND PROSPECTS FOR DEVELOPMENT

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 13 Jul 82) pp 7-11

[Article by Doctor of Economic Sciences Valentin Arsent'yevich Zgurskiy, Chairman of Ispolkom of Kiev Gorsovet, Candidate of Technical Sciences Yuriy Iosifovich Oprisko, Scientific Production Association Gorsistemotekhnika, Kiev, and Candidate of Technical Sciences Marat Alikovich Guriyev, Scientific Production Association Gorsistemotekhnika, Kiev]

[Text] Introduction. The Kiev Automated Control System Complex was developed within the framework of the program for automation of management of the country's large cities, begun during the 10th Five-Year Plan and which includes ASU for Moscow, Leningrad, Kiev, Yerevan, Tashkent and so on.

The innovator and scientific supervisor of the program was Academician V. M. Glushkov. The USSR State Committee for Science and Technology is coordinating the work.

Investigation toward development of city ASU complexes are directed toward phase arrangement of city management on the basis of extensive use of computers and result in introduction of modern control technology in which the use of economic and mathematical methods becomes an inseparable part of the activity of the management apparatus. The city ASU complex is a multilevel hierarchical level system that includes production process control systems, automated enterprise and association management systems and sector (within a city), intersector and citywide ASU [1, 2].

A city is regarded as a control entity in the form of an aggregate of four components: the population of the city, the urban territory, the town-forming base and the service sphere.

The general goals of developing a city ASU include improving the methods of management of the town-forming base and the service sphere, introduction of efficient methods of preparation and adoption of objective management decisions for dynamic and proportional development of the city and of its sectors, systems analysis and forecasting of trends of socioeconomic development of the region and improvement of substantiation, development and realization of complex plans for socioeconomic development of cities.

These goals are made more specific with regard to the specifics of development of each city.

The high growth rates of the city population during the past decade, which comprise an average of more than 2.5 percent annually, exceeding similar indices for Moscow and Leningrad, are considered as the control entity as the main feature of Kiev in development of the Kiev ASU. The accelerated growth of the city is the result of using intensive methods of development of enterprises and organizations of the town-forming base with the accompanying increase of the number of workers at these enterprises and an increase of the error between the indices of the production and non-production sphere and deterioration of the city ecology.

In this regard, the main goal of developing the Kiev ASU is formulated as guaranteeing the scientific-methodical and organizational-technical base for fulfillment of the three large interrelated functions of management of Kiev:

complex social development of the population;

territorial intensification and efficient functioning of enterprises and organizations of the town-forming base;

guarantee of social and ecological balance of sector development on the city territory.

The first of these functions is performed mainly by optimum development of the service sphere, which is a complex and dynamic mechanism in a large city. For example, approximately 330,000 persons are involved in the service sphere in Kiev, which corresponds approximately to the number of those involved in the sector of republic subordination.

Thus, one of the goals of the Kiev ASU is related to improvement and automation of city departmental management and is adequate to the goals of developing sector ASU. However, the main characteristics of the system are determined by the functions of territorial intensification, innovation and balancing of the city.

It is important in this regard to formulate the general objectively necessary task of territorial management of the city with embodiment in it of the normative content determined by the dynamics of the managed entity according to V. M. Glushkov [3].

The objectively required task of city management. It is known that the objectively necessary task for complex production systems is that of efficient synchronization of production processes and that the quality of production management is determined in the final analysis by the accuracy achieved in solving this problem. When converting to regional systems, one must take into account the two most important special factors that distinguish these systems from production factors: the population and territory of the region.

It has been suggested that one use the criterion of minimum expenditures of the useful time of the population as a version of the criterion of optimization of city development for thorough consideration of these factors in town-building theory [4]. In this case the sum of the useful time includes working time expended by the population for production of a product, work and services and part of the free time required to receive the services in the service sphere.

Minimization of the time in production and in servicing with the nomenclature of products, work and services established for the city, if recreational services are included in the services, essentially corresponds to all of the above-named main goals of city management. Therefore, rational minimization of the total useful time of the city population can be regarded as the objectively necessary problem of city management in the more general postulation and that calculated for the long term.

Direct solution of this problem in city-building practice is being delayed by the complexity of measurement and accounting of the actual time expenditures of the population.

However, this difficulty can be avoided in the practice of ASU development by using the well-known principle of managment by deviation: in management practice it is sufficient to estimate what the total saving of useful time for the population is provided in the final analysis by rationalization or optimization of one or another class of solutions in city management.

Making this task more specific for the individual components of time expenditures permits a corresponding refinement of the formulation of the objectively necessary management. For example, a significant part of the useful time of the population comprises transport time required to move from the place of residence to places of work or recreation and return. Transport time characterizes the so-called obstacle nature of the space of a territorial region and is an important index of the level of socioeconomic development of the territory. Continuing in similar fashion with production systems, one can compare the territorial objectively necessary task of congruence--efficient combination of processes and production and consumer entities in space--to the objectively necessary production task (synchronization or efficient combination of the result of production operations and processes in time).

The use of the theory of efficient minimization of the time of the city population classifies the development and monitoring of fulfillment of the complex plan for socioeconomic development of a city and at the same time permits one to pose new tasks of territorial management. An example of these tasks may be the exchange of leased sites between organizations having wide dispersion. The result of the exchange is concentration of the sites of each of these organizations within one or two rayons of the city and the corresponding saving of work time expended on travel between sites.

An important component of useful time is the time expended on city management. It is natural that the structure of using this time is of primary interest in development of city ASU. Investigations showed that up to 40 percent of the

time is expended in the total budget of management time on data gathering and less than 20 percent is expended on preparation of decisions and making decisions. In this regard the phase nature of developing the Kiev ASU envisions advanced development of automated information retrieval systems with subsequent integration of these systems in the citywide data bank and solution of optimization problems of planning and management on this basis.

Organization of development and introduction of the Kiev ASU. The main developer of the Kiev ASU is the Scientific Production Association Gorsistemotekhnika, directly subordinate to the Kiev Gorispolkom and having within it a scientific research and planning-design institute, information computer centers and also a special design-production office of nonstandard means of mechanization and automation of the urban economy. Due to the availability of computer centers, the association provides for introduction and operation of developed problems. A permanent committee of deputies on problems of scientific and technical progress works to manage the development of the Kiev ASU within the Kiev city Soviet.

The scale of the problems of management of Kiev required recruitment of many scientific research and planning organizations of the city to solve them, besides the main developer. In this regard the more important investigations are included in the specific complex urban program (KTsP of the Kiev ASU), confirmed by the decision of the bureau of the Kiev party Gorkom. A total of 29 enterprises and organizations is participating in fulfilling the tasks of the KTsP of the Kiev ASU. Among this number are Ukrniinzhproyekt [not further identified], the production-operational association Kievenergo and UGPI [Ulyanovsk State Pedagogical Institute] Tyzhpromavtomatika, which develop production process ASU for gas supply, electric supply, heat supply and water supply for the city, respectively.

Institutes of the Ukrainian SSR Academy of Sciences, primarily the Institute of Cybernetics and the Institute of Economics, Government and Law, are widely participating in the investigations on the Kiev ASU. For example, transfer of a number of promising applied program packs for data base management to the Institute of Cybernetics at the NPO Gorsistemotekhnika permitted a significant reduction of expenditures on the general systems software of the Kiev ASU.

The Kiev ASU is being developed in cooperation with the main organization for development of city management ASU in the USSR--the NPO Moscow ASU, and also the NTPO [Scientific and Technical Production Association] Lensistemotekhnika and other organizations of this profile.

The first unit of the Kiev ASU. The first unit of the Kiev ASU complex was put into operation in 1980 and includes 16 subsystems that combine more than 160 tasks of all levels of the city management hierarchy.

The ASU Gorsovet, which includes an automated complex of the information-dispatcher station of the ispolkom and monitoring and analytical systems for working with appeals of the citizens and execution of the decisions of the city Soviet and of its executive committee, is operated on the citywide level.

Various means of organizational equipment, including movie and slide projectors, television monitors, regulators and information boards, are used effectively as the information-dispatcher station. Local information reference systems that employ alphanumeric and color graphical displays are used for operational informing of the managers of the ispolkom and workers of the apparatus.

The use of video equipment in combination with operational data prepared on a computer and depicted on displays increases the accuracy of analysis and decision-making by the ispolkom of the city Soviet.

The use of computers for monitoring and multi-aspect analysis of working with citizen appeals yields a discernible social effect. The number of these appeals comprises tens of thousands annually and manual methods of checking and analysis are ineffective.

Departmental problems of planning and bookkeeping and accounting in organization of the city economy are solved at the intersector level. Moreover, a number of problems on management of city rayons is solved. They are mainly problems of operational and routine monitoring of the activity of the enterprises of the rayon and monitoring and analysis of the quality of products produced.

A number of analytical accounting problems for organizational-party and ideological work is solved for the party raykoms.

Data processing systems and management of the housing economy have achieved the greatest development in the first unit of the Kiev ASU at the sector level. Thus, a data bank of apartment buildings of local Soviets has been developed and is operating for the city housing administration and an automated information reference system for priority to receive housing has been developed and is operating for the city accounting and housing distribution administration. Lists of the monthly Bulletin for Housing Exchange have been published and lists for individual requests for exchange have also been formulated for a number of years on a computer for the city office for apartment exchange.

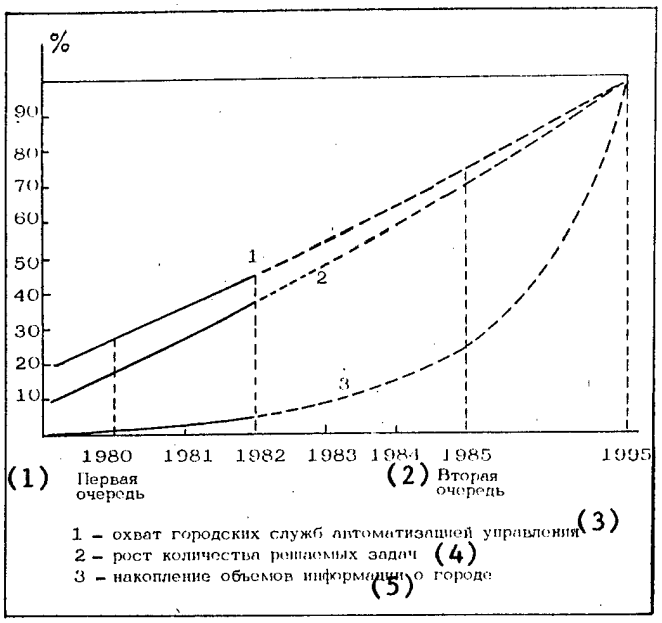
Many problems in management of the pharmaceutical industry, urban electric transport and in development and equalization by rayons of the city commercial network have been developed among other sector systems. Despite the predominance of accounting-statistical and checking-analytical problems in the first unit of the Kiev ASU, there is no doubt of the effectiveness of using computers to solve them.

For example, more than 600 different lists and analytical tables, which is equivalent to the manual work of 15 persons for a year, were printed on a computer for rayon departments for accounting and distribution of housing during 1981. A return to manual methods of checking and analysis of the appeals of citizens would result in an increase of 100 or more persons on the staff of inspectors in the city and rayon executive committees.

The confirmed saving from introduction of the first unit of the Kiev ASU exceeded 800,000 rubles. The social effect from increasing the accuracy of decisions in city management and an increase in the clearness and smoothness of functioning of the sectors and enterprises of the service sphere must also be taken into account.

Thus, the first unit of the Kiev ASU encompasses a wide range of city management problems on different levels, which in many cases "grew into" management technology and create a rather firm base for further development of the system.

At the same time, these problems cannot be overestimated for long-term development of the Kiev ASU. It should be stated that most of the introduced problems still retain the nature of departmental administration and the adopted methods of batch processing of data limit automation of the functions of operational planning and management. The support equipment of the Kiev ASU has been insufficiently developed and problems of training of user personnel have not been solved.



Key:
1. First unit
2. Second unit
3. Inclusion of city management automation services
4. Increase of number of problems to be solved
5. Accumulation of volumes of information about city

Development of the complex of Kiev ASU. According to the theory of the minimum expenditures of useful time of the population and according to the objectively necessary problem of management outlined above, the basic directions for development of the Kiev ASU are related to expansion of the range of

problems of territorial matching of sector interests in the city territory and (simultaneously) are related to increasing the efficiency of the methods and means of all types of service to the population on the basis of development of a city network of automated information dispatcher systems (AIDS).

An increase in the volumes of information to be processed about a territory, basic funds and population of the city, expansion of the scope of automation of city service management and a significant increase in the number of problems to be solved are provided.

The dynamics of variation of some generalized indicators of development of the Kiev ASU are presented in the Figure.

A number of problems of territorial planning of the development of Kiev, in which significant expansion of the information service of departments of the party gorkom, gorispolkom and city planning committee is provided at the phases of formation and confirmation of the draft of the territorial plan, will be introduced during 1982-1983.

The automated system for coordination of excavations of city territory to be developed, which will solve the problem of reducing repeat excavations and will guarantee efficient coordination of the deadlines of restoration, construction and repair work on all types of city engineering supply lines, can also be included among systems that are representative of the direction of territorial coordination. The basis of the system is the automated data bank on urban roads (ABD Dorogi), effective by the 1,500th anniversary of the founding of Kiev.

Introduction of the first two AIDS: Gostinitsa and Skoraya pomoshch', will be timed to the city anniversary simultaneously with the ABD Dorogi.

The AIDS Gostinitsa made it possible to centralize and bring order to the processes of distributing advance applications for residing in hotels. The process of distributions of calls for first aid between substations and brigades has been automated and information service of the population on problems of emergency hospitalization has been improved in the AIDS Skoraya pomoshch'.

Development of support equipment. Development of systems engineering support of the Kiev ASU complex is related to a number of organizational and technical difficulties, determined by the interdepartmental and intersector nature of the problems to be solved, by the existing restrictions on teleprocessing processes and by the nonsolution of a number of organizational and legal problems of automation of territorial management.

A synthetic approach to development of the system [3] that permits development of functional management complexes at advanced rates and then to accomplish information, software-hardware and organizational-legal interrelation of them in this regard during development of the Kiev ASU.

The main problems of information support of the Kiev ASU include those of formation of lists of the basic nomenclature of territorial management

entities, design of systems for management of basic nomenclatures, design of information languages and creation of conditions for describing the subject area of city management.

The automated system for management of union-wide classifiers of technical and economic information, which is being supplemented gradually by departmental classifiers of territorial-oriented information, is the basis of information support. The problem of this supplementation reflects the specifics of the Kiev ASU compared to sector ASU and is caused by the need for detailed classification of the subject area of city management in terms of coordinate systems and space metrics of the city territory.

The need to solve the problem is obvious from the fact that there are few enterprises and organizations in Kiev, as in any other large city, in which rayons of party subordination, territorial disposition and administrative affiliation do not coincide. These facts are the consequence of establishment of the national economy and do not contradict the existing legislation, but they present considerable difficulties for the analytical balances and calculations for the city as a whole and by rayons. This results in a need to use several schemes for classification of rayon division and so on in the Kiev ASU.

An important problem of information support of the Kiev ASU is theoretical analysis and practical solution of the principles of the relationship of three information processes in city management--basic data processing for management of the Kiev national economic complex, preparation of analytical data for party management of socioeconomic development of the city and the process of expedient informing of the city population.

Among other problems should be distinguished normative-information support of processes of territorial coordination of sector plans: regulation of these processes requires creation of a special class of regional norms that define both the degree of functional effectiveness of entities of the town-forming base in a city and the degree of their participation in development of the service sphere.

The complexity of solving the problems of information support of the Kiev ASU was determined by the significant volumes of data required to solve the problems of city management. The volume of these data for problems of the first unit comprised 29 Mbyte in 1980, up to 60 Mbyte in 1982 and it is planned to have approximately 250 Mbyte of conditionally permanent data in the computer memory by 1985, including the fact that data on the basic funds of the city economy will comprise approximately 125 Mbyte and data on the population will comprise 55 Mbyte. According to preliminary estimates, a total of 4.6 Gbyte of conditionally permanent information must be entered in the computer for future solution of the objectively required problem of Kiev city management.

The number of main problems for development of algorithmic support should include:

development of methods of using existing man-machine optimization equipment, accounting, analysis and checking of planning and economic

indicators of the functioning of national economic entities oriented toward efficient use of paperless technology under conditions of city management;

development of special methods of territorial (spatial) optimization of the disposition of city entities, including entities of municipal housing management, transport, commerce and so on;

the use of methods of simulation of territorial systems in Kiev for formation of long-term forecasts and optimization of future planning.

The use of postulations of problems of organizational management and ASUTP used in machine building sectors of industry in the sphere of management of sectors of the national economy is an important methodological direction.

The YeS-1045, YeS-1033 and YeS-1022 computers, recently supplemented by computers of the SM series, comprise the basis of the hardware for the Kiev ASU.

It is planned to create a two-level computer complex with city-rayon territorial hierarchy. In this case the GIVTsKP [collective-use main sector computing and data processing center] of the gorispolkom will take on itself the main load in solution of citywide and intersector problems and the rayon VTsKP, created in phases, will solve problems of the ASU rayon and of sector problems uniformly distributed between them.

Introduction of the GIVTs [main sector computing and data processing center] based on the YeS-1045 computer is planned during 1982-1983 and development of a network of rayond VTsKP based on the computer capacities of the NPO Gorsistemotekhnika with phase recruitment of reserves of the computer capacities of sector VTs locate din the rayons is planned during 1983-1987.

An important and more laborious component of the hardware complex of the Kiev ASU is the system of message switching centers that are organizationally and territorially combined with corresponding VTs. According to the general scheme for development of the hardware complex of the Kiev ASU, creation of a complete two-level system of message switching centers is planned for completion during the 13th-14th Five-Year Plans.

Software is being developed for the Kiev ASU on the basis of extensive use of SUBD [database control system]. Whereas only two SUBD packs (BANK DOS and NSI-1-DOS) were used in the first unit of the complex, OKA, SPEKTR, INES, PAL'MA, DIAMS and other packs will be introduced in the second unit.

An important feature in development of all components of systems engineering support of the Kiev ASU and of the complex of systems as a whole is the requirement of subsequent copying of the functional management complexes in the large cities of the Ukraine, since the Kiev ASU is being developed as the main territorial link of the Ukrainian SSR RASU [Republic automated control system].

BIBLIOGRAPHY

1. Myasnikov, V. A., "Prospects for Complex Automation of Management of the City Economy," in "Upravleniye bol'shim gorodom: Tezisy dokladov 1-y Vsesoyuznoy konferentsii" [Management of a Large City: Report Topics of the First All-Union Conference], Moscow, NPO ASU Moskva, 1981.

2. Cherkasov, Yu. M., "Methodology and Organization of Creation and Development of the Complex of Interconnected ASU of a Large City," Loc. cit.

3. Glushkov, V. M., "Vvedeniye v ASU" [Introduction to Automated Control Systems], Kiev, Tekhnika, 1974.

4. Uspenskiy, S. V., "On the Problem of Selecting the Criterion of Optimum Organization of a City," in "Planirovaniye sotsial'nogo razvitiya gorodov" [Planning the Social Development of Cities], Moscow, 1973.

6521
CSO: 1863/72

UDC 681.327.8

ORGANIZATION OF DATA EXCHANGE BETWEEN COMPUTER AND REMOTE USERS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 24 Nov 81) pp 25-28

[Article by Candidate of Physicomathematical Sciences Petr Matsovich Ivanov, engineer Vladimir Titovich Kerefov and engineer Nikolay Alekseyevich Tkhishev, Kabardino-Balkar Branch of All-Union Production and Technological Association Rossel'khoztekhsistema ASU, Nalchik]

[Text] As is known, development of modern information computer systems and automated control systems (ASU) is accompanied by the rapid growth of the number of remote users of computers, the volumes of data transmitted by them and the related increase of the complexity and cost of the data transmission system (SPD). According to some estimates [1, 2], the fraction of expenditures for development of SPD has already reached 40-60 percent of the expenditures for development of the entire ASU and continues to increase. Under these conditions, the problem of increasing the efficiency of using communications channels and channelling equipment acquires great significance. The main method of solving this problem at present is organization of collective-use channels by using remote data concentrators and multistation connection of users to communications lines [2, 3]. However, the data exchange control procedures are usually complicated in this case, as a result of which the amount of service data (control, address and so on) transmitted and the equipment used increases and part of the calculating resources of the computer is allocated for control of the SPD.

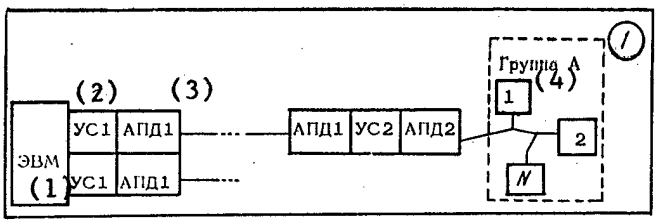
At the same time, territorial grouping is typical for most remote users of computers, i.e., they are arranged so that they can be combined into groups, the distance from to the computer is considerably greater than the distance between the users within the group themselves. This characteristic can now be used successfully to combine the advantages of multiterminal systems and systems with data concentration and design on this basis of inexpensive multiterminal SPD with high utilization factor of extended communications channels when output of a wide nomenclature of modern data transmission devices has been organized within the YeS EVM [Unified computer system] and ASVT-M [modular system of computer equipment on microelectronics base].

One of the possible versions of this SPD is explained by the diagram shown in Figure 1. The user stations (AP) 1...N, which comprise a single group (for

example, located within one large industrial enterprise, rayon or city), are connected by a mainline to a single local communications line. The local communications line is controlled by a central station (TsS) for APD2 data transmission equipment and US2 integration device. Data on the users are transmitted to the TsS and then through a common extended channel equipped with APD-1 data transmission hardware in the computer. Data is transmitted from the computer to users in reverse order. The US1 and US2 integration devices guarantee intraction of the APD1 with the computer [5, 6] and the APD2 TsS, respectively.

Hardware realization of the multistation communications channel control procedures in the APD2 and US2 in this system permits one to reduce to a minimum the volume of service (control) data transmitted over the extended communications channel. If, moreover, an APD that carries out the control procedure with respect to hardware and that increases the reliability of data transmission over the extended section of the SPD, i.e., that operates on the "clean" tape principle, is used as APD1, then the computer can be completely freed of SPD control functions. The service part of the message transmitted from the computer (or to the computer) will contain only the number (address) of the user from which (to which) this message is transmitted.

The number of users connected to a multistation communications lines, the order of the polling and the ratio of the data transmission speeds in different sections of the SPD are arbitrary in the general case. But, since the users utilize a time-shared local communications line, the data transmission speed on all sections of the SPD can be set identical to simplify the US2 and the user equipment. For this purpose, one can limit oneself to two simple user polling disciplines: nonpriority (circular polling) and priority (for example, polling with return to the beginning of the list after servicing of the user's request is completed and so on).



Structure of SPD for Data Exchange Between Computer and Group of Users

Key:
1. Computer
2. Integration device
3. Data transmission equipment
4. Group

The systems programs can easily be designed by using standard programs for control of dialogue peripheral devices (displays and so on).

The speed and reliability of data transmission and also requirements on communications lines in this SPD are determined by the corresponding characteristics of the data transmission equipment used. And such SPD characteristics as the utilization factor of the extended communications channel and the average

message delay time can be determined when considering the model of a system by using queueing theory [7]. In the given case this model is a queueing system with a single service device to whose input the flow of declarations for message transmission is fed. The moments of arrival of the declarations are described with sufficient accuracy by Poisson distribution with mean value of the intensity of declarations per second, while the message length is characterized by exponential distribution with mean value of k characters per message. The declaration arriving at a moment when the service device is engaged gets in line and awaits servicing.

According to [8], the utilization factor in this system is determined by the expression

$$y = \lambda x = \lambda \frac{k}{c},$$

where x is the average transmission time of one message (s) and c is the throughput of the extended communications channel (characters per second).

The average message delay time in the system consists of the time of waiting for servicing W in the queue and the message transmission time:

$$T = x + W = x + \frac{yx}{1-y}.$$

If remote users of the computer are working in the dialogue mode, then most declarations for message transmission cannot exceed the number of users. In this case [9], the average delay time will be equal to

$$T = x + \frac{\sum_{i=2}^{N} (i-1)(\lambda x)^i}{\lambda \sum_{i=0}^{N} \frac{(\lambda x)^i}{(N-i)!}}.$$

The probability that a declaration arriving in the system will await servicing is equal to

$$P = \frac{\sum_{i=2}^{N} \frac{1}{(N-i)!} (\lambda x)^i}{\sum_{i=0}^{N} \frac{1}{(N-i)!} (\lambda x)^i}.$$

The mean value of the intensity of arrival of declarations to the servicing device is determined as the sum of the mean values of the intensity of issue of declarations for message transmission by each of the users:

$$\lambda = \sum_{i=1}^{N} \lambda_i .$$

The average time the user waits for an answer to his message in the dialogue system is equal to

$$T_{ож} = 2T + t,$$

where t is the time required to prepare the answering message in the computer.

An SPD using the described method of collective-user channel organization [10] has been created in the Kabardino-Balkar ASSR within the framework of developing an ASU for an agroindustrial complex of oblast (ASSR or kray) level. The system guarantees operational data exchange between the YeS 1022 computer, located at the republic IVTs [information computer center] and the users of the rayon center or of a large enterprise of oblast (ASSR or kray) subordination (repair plant, supply base and so on).

APT-MA-TF equipment is used as the APD1 and APD-MPP equipment is used as the APD2. Each user is supplied with an AP that includes, besides the APD-MPP user station, an Iskra-2302 electronic bookkeeping-billing machine and Videoton-340 (VTA-2000) display. The user stations are connected to the central station of the APD2 by physical or by segregated city telephone lines (control ATS [automatic telephone exchange]), the length of which should not exceed 7 km at a transmission speed of 65 characters per second and 5 km at a speed of 130 characters per second. Data is exchanged between the group of users and the computer over a segregated long-distance telephone channel at a speed of 65 or 130 characters per second (600-1,200 baud)

The message transmitted to the computer (from the computer) over the extended communications channel contains the address character of the user from whom (to whom) data are transmitted and the data itself. The address character is formed during message transmission to the computer by hardware in the US2 and by software during transmission from the computer to the US2. The device described in [5] is used as US1 and the device specially developed for this system and that performs the following functions is used as US2:

conversion of APD-MA-TF and APD-MPP signals and matching of them;

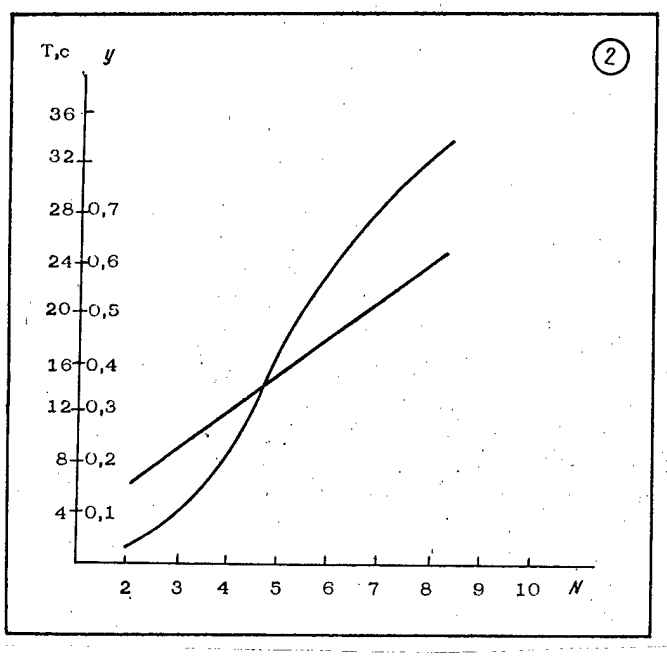
control of AP polling;

shaping of address characters during data transmission from the AP to the computer;

decoding of address characters during data transmission from the computer to the AP;

caching of the transmitted data.

The experience of operating user stations at one of the sectors of the agro-industrial complex--Goskomsel'khoztekhnika [State Committee for Supply of Production Equipment for Agriculture] [11], connected to the computer by private communications lines and those operating in dialogue mode, showed that the average message length is K = 490 character, while the average intensity of arrival of requests for message transmission is equal to = 0.02 1/s. the dependence of the average delay time T and utilization factor of the extended communications channel y on the number of users N in a group, found by the expressions presented above with regard to these statistical data at transmission speed of 135 characters per second (1,200 bit/s), is shown in Figure 2. As can be seen from Figure 2, the load of the extended communications channel increases as the number of users increases. However, the message delay time also increases rapidly and may assume larger values. As a result, it may be feasible to combine a smaller number of users into a group than hardware permits when high operational interaction of users and the computer is required. This number should be determined in each specific case with regard to the nature of the problems which the users will solve.



Dependence of y and T on Number of Remote Users N

A diagram of the algorithm of systems operation is presented in Figure 3. The procedures which are realized by hardware in APD1 and APD2 and by user station equipment are not shown on the diagram to simplify the description.

In the initial state (1A, 1B), there is no request in the system at all for data transmission, all the communications channels are free and the US2 is set to the polling mode (1C-1E). The US2 forms the address code of the user station during polling according to the polling discipline, while the APD2 calls the indicated user station. If there is no request at the user station for transmission (1E), then the US 2 transmits the next address code and everything is repeated.

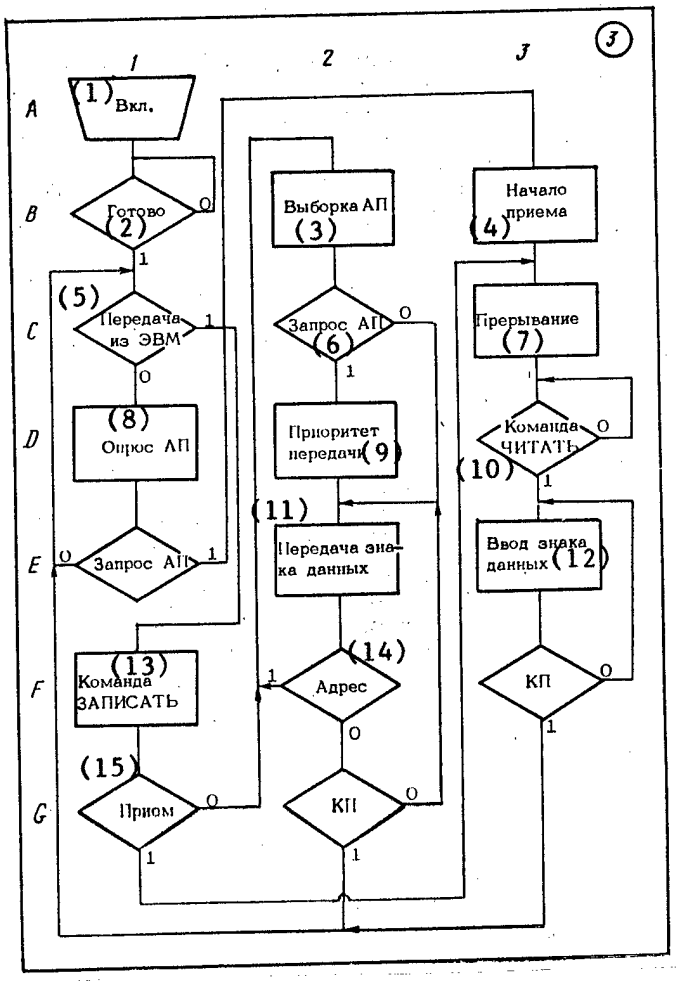


Diagram of Algorithm For System Operation

Key:

1. On
2. Ready
3. Selection of user station

[Key continued on following page]

4. Beginning of reception
5. Transmission from computer
6. Request of user station

[Key continued from preceding page]

7. Interrupt
8. Polling of user station
9. Tranmission priority
10. Instruction READ
11. Transmission of data character
12. Entry of data character
13. Instruction WRITE
14. Address
15. Reception

The US2 permits polling by two disciplines:

circular polling with continuation (polling is restored from the same point in the polling list after the incoming request has been serviced);

circular polling with return to the beginning of the list.

There is the capability of excluding any user station from the polling list which is required during adjusting operations at the user station, when the user station is operating in the autonomous mode or when it is absent altogther.

If it turns out during call of the next user station in the polling cycle that a request has been issued for transmission, polling is stopped (1E), the US2 forms the character of this AP and transmission of the message to the computer begins. At the signal to begin reception, transmitted at the junction of the APD1 connected to the computer (3B), the US1 device forms an interrupt request (3C). In response to this request, the computer should issue the READ instruction (3D) by which the message is entered in the computer (3E). Upon completion of message transmission (3F), the system returns to the initial state.

If it is necessary to transmit a message from the computer to any user station (1C), the computer issues the WRITE instruction (1F) to the US1 by which the APD1 connected to the US1 converts to the transmission mode and begins transmission of the message. At the signal to begin reception from the US2, the APD1 stops polling and message reception begins. Having decoded the address character, the US2 selects the user station (2B), supplying the address code of the user station to the APD2 junction and the request for transmission. If there is no counter request at this user station (2C), then the message is transmitted to the recipient (2E). If a counter request is transmitted to the user station, the US2 switches on the priority transmission mode of APD2 (2D), the request of the user station is disregarded, the equipment of the user station is switched to the receive mode and the message is transmitted to the recipient. The system returns to the initial state upon completion of message transmission (2G).

If several messages are transmitted one after the other from the computer, the US2 identifies the address character of the user station standing at the beginning of the new message, stops transmission of the previous message and selects the user station according to the decoded address and the next message is transmitted to the recipient.

A situation is possible when the computer issues the WRITE instruction to the US1 at a moment when the APD1 is in the receive mode, but the begin reception signal (1G) has not yet been transmitted to the junction. In this case the US1 forms the attention interrupt (3C), in response to which the computer should enter the message coming from the communications channel (3D, 3E or 3F) and the WRITE instruction is then repeated.

Operation of the system over a number of years confirmed the results of calculations, which determines the feasibility of using the described organization of remote user-computer interaction.

BIBLIOGRAPHY

1. Gerasimenko, A. P. and B. V. Nikolayev, "Abonentskiye ustroystva dlya svyazi s EVM" [User Devices for Communicating with Computer], Moscow, Svyaz', 1979.

2. Martin, J., "Seti svyazi i EVM" [Communications Networks and Computers], Part 1, Moscow, Svyaz', 1974.

3. Davis, D. and D. Barber, "Seti svyazi dlya vychislitel'nykh mashin" [Communications Networks for Computers], Moscow, Mir, 1976.

4. "Firmware of Real-Time Automated Control Systems," "Materialy seminara" [Proceedings of a Seminar], Moscow, MDNTP, 1978.

5. Ivanov, P. M., V. T. Kerefov, P. G. Teslya et al, "Integration of the APD-MA to the Unified Computer System," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 3, 1980.

6. Korchinskiy, A. I., A. F. Bulychev, A. R. Golubtsov et al, "Multipleksory peredachi dannykh" [Data Transmission Multiplexers], Moscow, Energiya, 1980.

7. Saati, G. L., "Elementy teorii massovogo obsluzhivaniya i yeye primeneniya" [Elements of Queueing Theory and Its Applications], Moscow, Sovetskaya radio, 1971.

8. Neyman, V. I. "Computer Communications Networks," ITOGI NAUKI I TEKHNIKI, SERIYA ELEKTROSVYAZ', Vol 9, 1978.

9. Malinovskiy, S. T., "Seti i sistemy peredachi diskretnoy informatsii i ASU" [Digital Data Transmission Networks and Systems and Automated Control Systems], Moscow, Svyaz', 1979.

10. Ivanov, P. M., "ASU proizvodstvenno-tekhnicheskim obespecheniyem sel'skogo khozyaystva" [Automated Control Systems for Production and Technical Support of Agriculture], Nalchik-El'brus, 1981.

11. Ivanov, P. M., I. I. Gridyakin, N. A. Tkhishev et al, "The APD-MA--Videoton-340--Iskra-2302 Terminal," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 2, 1980

6521
CSO: 1863/72

UDC 621.771:65.011.56

CHARACTERISTICS OF USING CONTROL COMPUTER COMPLEXES IN AUTOMATED PRODUCTION PROCESS CONTROL SYSTEMS IN WIDE-STRIP ROLLING MILLS WITH PROCESSOR CONTROL

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 (manuscript received 1 Jun 82) pp 97-98

[Article by Doctor of Technical Sciences Ivan Nikolayevich Bogayenko, Candidate of Technical Sciences Anatoliy Nikolayevich Volynskiy and Candidate of Technical Sciences Anatoliy Ivanovich Sbitnev, Institute of Automation, Kiev]

[Text] Investigations were begun in the early 1970s in our country (including at the Kiev Institute of Automation) on development of ASUTP [Automated production process control system] at wide-strip thin-sheet rolling mills. The theoretical basis for developing these systems were fundamental investigations in the field of automated control systems, carried out under the supervision of Academician V. M. Glushkov. Investigations in the field of rolling theory, conducted under the supervision of Academician A. I. Tselikov, played an important role in development of these ASU.

The digital-analog process of rolling at a wide-strip mill is characterized by a large number of adjustable coordinates, the need to take into account the control effects of data during calculation due to the significant number of meters (up to 120) and the variable transport delay and the presence of more than 20 interconnected control circuits of the main production parameters with variable transfer factors of the entity.

Processor control systems based on third-generation control minicomputers were developed and introduced at a number of continuous hot and cold strip rolling mills during the mid-1970s at the Kiev Institute of Automation under the scientific supervision of Academician of the Ukrainian SSR Academy of Sciences B. B. Timofeyev and his students Yu. P. Bobranitskiy, V. I. Vasichkin, N. G. Lysenkov and Ye. V. Leonidov-Kanevskiy.

The characteristics of these systems include the following:

organization of control circuits (using difference control equations), optimization circuits and adaptation of models on the program level;

extensive use of standard averaging and approximation procedures by splines for the output characteristics of production parameter meters;

the presence of display type devices for operational imaging of production data, including that in the form of graphs and histograms;

the presence of problems related to gathering, processing, display and registration of the technical and economic operating indices of the production equipment;

extensive capabilities of diagnosing the entity, control systems and meters.

One of the first direct processor control systems was the ASU for accelerated cooling of strips on the hot-rolling mill 200 of the Novolipetsk Metallurgical Plant in 1977. The seventh basic M6000 control computer complex with memory capacity of 32K was used in the system.

Data from 70 meters (strip temperature, rolling rate, water flow rate and so on) were entered in the control computer complex for control. The control computer complex issued 40 analog and 80 digital control actions. A total of 12 control circuits was realized at the program level, of which four were closed, four were open and four were model adaptation.

The system provided regulation of two parameters--the cooling rate and temperature of winding the strip that determine the product quality for mechanical properties.

An automated strip-width control system and also a system for accounting for the rolled product in theoretical weight, accounting for the rolling rate and idle time of the rolling mill were introduced at the same mill on the second M6000 computer. The automated strip-width control system was designed to determine the maximum negative tolerance in thickness for each rolled strip, adjustment of the clamp screws of the finishing group stands and also for stabilizing the thickness of the strip at the mill output to achieve the minimum possible variation of thickness with respect to the established value.

The characteristic feature of the given system is that control of such crucial mechanisms as the clamping devices of the finishing stands is realized on the basis of the M6000 control computer. This places very high requirements on the reliability of the computer equipment and programs of the system. Devices for non-impact switching to the local control mode with the control computer shut down were developed especially for the system.

The special systems programs utilize different methods of recurrent evaluation of the parameters of random processes and stochastic approximation for adaptation of the control models.

The systems programs include a synchronization unit, strip tracking unit, database management unit, thickness control unit, unit for determining the theoretical weight of the rollers and a unit for taking into account the productivity of the mill.

To simplify the interprocessor communications between two complexes, the ASUTP functions with two-level structure of an ASUTP with two UVK [control computer complex] (ASUTP of the mill 2500 of the Magnitogorsk Metallurgical Combine) are divided into two parts and execution of them is time-shared.

The first part is the automated subsystem for initial adjustment of the position and speed mode of the mechanisms, which performs its control functions during the pause between rolling of strips, while it gathers and processes technological, engineering and production data during the rolling process. Control is maintained by acting on the local digital systems of the position of the clamping screws, straightedges and local digital control system of the speed modes of the finishing stands.

The second part--the automated subsystem of multiparameter processor control of production parameters--performs control functions, beginning at the moment the strip enters the finishing stand, with respect to the working point of the rolling process.

Control is accomplished by signals of production sensors installed at the input to the finishing stands and at the output from it. The control process in each rolling stand and in each space between stands is performed by using autonomous controllers.

Control is maintained in the iteration relay mode, determined by the fact that the control entity--the finishing stands--contains considerable transport delays determined by the movement of metal from the point of control to the point of measurement of the controlled parameter. These delays comprise 1-10 seconds as a function of the type of parameter and the rolling speed.

There are two types of control circuits: for deviations--signals generated on the basis of data from sensors installed beyond the finishing stands and by perturbations--signals generated at the input to the finishing stands. A delay of measurement after issue of the control action is typical for the deviation control circuits and prohibition of measurement prior to introduction of the control action, on the contrary, is typical for perturbation control circuits. Synchronization of issue of control at the rate of motion of the metal to the control processing zone is important for the latter case.

Regulation is accomplished simultaneously for more than 10 autonomous control circuits with adaptation of control circuit coefficients. The longitudinal thickness, different transverse thickness of the strip, width of the strip, temperature of the end of rolling and winding temperature are regulated and the pressure in the last stands is stabilized. The characteristic feature of control is the communication of these circuits through the strip, which serves as a carrier of the control results from stand to stand in the form of deviation of the geometric dimensions of the strip or of the temperature mode of the metal to be rolled.

ASUTP using the M7000 control cmputer and microprogram automatons have been put into operation on continuous cold-rolling mills 1700 of the Karaganda Metallurgical Combine and the Zhdanov Metallurgical Plant imeni Il'ich.

The tasks of these ASU include calculation of the optimum adjustment of the rolling mill by the dynamic programming method, operational accounting for the status of the rollers with the capability of issuing a warning signal of the need for roll changing, processor control of the thickness of the strip in negative tolerance and gathering, processing, display and recording of technological and production data.

The main feature of the ASUTP programs at wide-strip rolling mills is the need to solve problems of synchronizing the calculating process with the production process; hence the compulsory presence of two functional components of special programs such as "Initial start" and "Emergency restart" and a subsystem for tracking the metal and also the need to supplement the staff operating systems with a developed synchronization system (from the initiative signal processing unit to the signal system).

Considerable simplification of the program's structure is achieved by using a database. The contradiction between the large volume of information stored for a long time and the rather high frequency of changing it (approximately 100 ms) is solved by using a distributed database: resident OZU [main memory] and resident disk.

The data taken from the sensors are identified by the tracking system (several moving strips are located simultaneously under different points of data removal on the mill and therefore the affiliation of each removed value of a specific strip must be determined) and are entered in the resident memory part, where the tasks that realize the control circuits and other functions of operational control are utilized. Prior to changing the old data, a check is then made for information content and the data are stored on disks. The main user of these data are the tasks of accounting and investigation of technology.

The resulting experience has fully confirmed the high efficiency of using control minicomputers for direct processor control of the production process of rolling. The following advantages can be noted in this case.

The universality of the firmware of control computers permits a 97 percent increase in the level of standardization and unification.

The periods and expenditures for systems design are reduced due to the fact that the sequential diagram of the stages of development is replaced by a sequential-parallel diagram in which the equipment can be ordered immediately following the draft, while delivery of the equipment can be combined with the contract and detail design.

Multicircuit and multicontact systems can be developed for complex entities that guarantee a high saving from introduction with comparatively short periods of return of investment.

Using the methodology of layer structure design of software, which combines the principles of functional-event decomposition with model-interface and

structural programming principles, guarantees flexibility and reliability of the system during introduction and operation by permitting phase adjustment of it and introduction (for individual problems) and modification of the system as the technology and composition of the equipment change.

6521
CSO: 1863/72

DEVELOPING TECHNOLOGY OF SCIENTIFIC RESEARCH AND INTRODUCTION OF RESULTS INTO NATIONAL ECONOMY

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 109-110

[Article by engineer Mikhail Antonovich Luchka, Computer Center of Institute of Applied Problems of Mechanics and Mathematics, Ukrainian SSR Academy of Sciences, Lvov, Doctor of Physicomathematical Sciences Mar'yan Vladislavovich Pashkovskiy, Lvov State University, and Candidate of Technical Sciences Vyacheslav Pavlovich Solov'yev, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] The first school-seminar "Development of Technology of Scientific Research and Introduction of Results into the National Economy," organized by the Presidium of the Ukrainian SSR Academy of Sciences, the Lvov Oblast Committee of the Ukrainian Communist Party jointly with the Council on Automation of Scientific Research attached to the Presidium of the USSR Academy of Sciences, the coordinating committee of the Ukrainian SSR Academy of Sciences "Systems Research of Organizational and Control Problems of Science and Technology," the Institute of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences, the Institute of Applied Problems of Mechanics and Mathematics, Ukrainian SSR Academy of Sciences, Lvov State University and the Drogobych Pedagogical Institute, was held at Drogobych (Lvov Oblast) from 23 through 29 October 1981.

A total of 112 specialists participated in the work of the seminar and representatives of 32 organizations from 11 cities of the country gave 25 plenary reports and 32 podium communications.

Academician of the Ukrainian SSR Academy of Sciences Y. S. Podstrigach opened the work of the school-seminar. Main attention was devoted in his report "The Program-Specific Approach to Management of Scientific and Technical Progress in the Region" was devoted to further increasing the efficiency of interdepartmental complexes and associations.

It was noted in the report of G. M. Dobrov (Kiev) "Timely Problems of Scientific and Technical Policy in Light of the Decisions of the 26th CPSU Congress" that development of the technology of scientific research is of increasing significance for improving the quality and increasing the efficiency of scientific research work according to the requirements advanced by the 26th CPSU Congress.

Questions of training personnel under conditions of interdepartmental scientific cooperation were raised in the report of Ye. I. Gladyshevskiy (Lvov) "Interdepartmental Scientific Contact--A Determining Factor of Personnel Training" on the example of the Lvov region as a whole and of the Lvov State University in particular.

The experience of organizing scientific research under conditions of training and production associations, which include Lvov and Kiev State Universities, was analyzed in the report of A. I. Zyubrik (Lvov). The operating experience of the associations shows that scientists of the higher school are capable of effectively managing applied research and of performing experimental-design work, as a result of which the deadlines for introduction of vuz developments into production are reduced appreciably.

Switching of the production and vuz sectors during the initial phase of formulation of scientific research was discussed in the report of I. V. Savitskiy, A. I. Zyubrik and D. L. Futorskiy (Lvov). A special department that recruits students to active fulfillment of the planned tasks of scientific developments was created at Lvov State University for specific training of highly qualified specialists for electronic technology. Participating in scientific research work, a future young specialist obtains the necessary experience of conducting independent research work, organizing skills and know-how in introduction of the results of work into production.

The structure of personnel training by the vuzes of the republic was analyzed in the report of M. V. Pashkovskiy and A. S. Krochuk (Lvov) "The Problem of Personnel During Formulation of Scientific Research." The deficiencies of the existing vuz system of training were revealed and methods of possible correction of them were indicated. Special attention was devoted in the report to the problem of training creative physicists and the practically total absence of a methodological base for training young specialists for work in academic institutions was noted.

The main properties of experimental data were analyzed, the principle of "distribution" of information circulating in the scientific experiment automation system was formulated and recommendations were given on this basis on organization of programs for automated data processing in the report of V. P. Solov'yev (Kiev) "General Principles of Automated Data Processing."

The main concepts of optimizing the perception of visual and aural information (based on the concepts of minimum data losses in the interactive mode) were outlined and some requirements on display devices were formulated in the report of B. N. Malinovskiy, Ya. N. Gnativ and M. A. Luchki (Kiev and Lvov) "Problems of the Psychological Comfort of the Investigator Under Automated Experiment Conditions."

The report of M. F. Derkach, V. N. Mazura and M. Ye. Chaban (Lvov) "Machine Recognition of Word-Organized Speech in the Specific Prediction Mode" was devoted to problems of "intellectualization" of man-computer dialogue.

Approaches to automated design of mathematical models for entities investigated experimentally were outlined in the reports of L. V. Matsevityy (Kiev) "On Automated Design of Mathematical Models from Experimental Results with the Entity to be Studied" and V. G. Grishko and V. A. Strel'chenko (Kiev) "Methodology of Research in Use of ASNI [Automated scientific research system] of Processes with Random Responses During Study of the Mechanical Properties of Materials and Structures."

The report of O. N. Romaniv (Lvov) "Problems of Development and Introduction of New Techniques Within the Interdepartmental Specific Scientific-Industrial Machine Building Complex" was devoted to problems of intensifying the introduction of scientific developments concerning development of new technologies and the specifics of the relationship of academic and sector organizations and enterprises in realizing the developments on technical re-equipping of sectors of the national economy was also considered.

The basic requirements on the firmware complex were formulated in the report of V. N. Starkov (Kiev) "Automated Problems Laboratory for Investigation of Solid-State Surfaces."

A number of general principles of development of automated problems-oriented laboratories were outlined in the reports of V. N. Korobeynikov (Kiev) "Automated Problems-Oriented Laboratory of Medical and Biological Profile" and of V. V. Syrov, V. A. Ivanova and A. I. Zharova (Kiev) "Analytical Complex of a Multiple-User Biochemical Laboratory." Moreover, data were presented in the indiated reports on the experience of operating the first prototypes of the developed systems.

Some generalizations of methods of designing automated scientific research and planning work systems were considered and practical recommendations based on many years of experience of the corresponding developments were given in the reports of A. N. Vystavkin, V. M. Vukolikov, A. Ya. Oleynikov and M. I. Pertsovskiy (Moscow) "Some Aspects of Developing Automated Scientific Research Systems" and of R. V. Rozhankovskiy and Yu. B. Kunovskiy (Lvov) "A Basic System of a Complex Scientific and Technical Program for Automation of Design Work."

The report of R. V. Magery, N. V. Odrekhivskiy and V. M. Shuplyak (Drogobych) "An Automatic Precision Temperature Measuring System" was devoted to a number of principle problems of temperature measurement under interference conditions and contained specific recommendations on technical realization of the measuring part of the system and on algorithms for processing the measuring data.

The podium communications mainly contained information on the experience of development and use of some specialized data processing devices and also special programs elements.

It was noted during discussion of the plenary reports and podium communications that an important factor for transforming the means and methods of automation of scientific research into a moving force for development of the technology of scientific research and introduction of it into the national economy

was coordination of work in computerization of the research process of the Council on Automation of Scientific Research attached to the Presidium of the Ukrainian SSR Academy of Sciences, and also the advances of the Western Scientific Center, Ukrainian SSR Academy of Sciences, in concentration of the efforts of academic, sector and vuz scientific organizations in complex solution of the more timely problems of introducing the advances of science into industrial production. However, the problem of reducing the deadlines of introduction of fundamental and applied research into the national economy to intensify production remains timely as before.

As a result of the discussions, the concept of the technology of scientific research was defined as a system of organizational and methodical measures that optimize the process of gaining new knowledge and converting it to components of social production, based on extensive analysis and generalization of the methodological principles of scientific creativity and a developed technical base.

It was noted in the decision of the school-seminar that the most timely problems for the near future include:

creation of the general concept of development of the technology of scientific research;

formation of models of the technology of scientific activity;

establishment of general terminology;

determination of measures of the level of research technology;

substantiation of the indices of proportionality for development of its main constituents and also the indices of the scientific and economic effectiveness of various types of technological solutions;

more active use of the arsenals of modern methods of applied and systems analysis and computer equipment in this field.

An excursion meeting at one of the enterprises of Drogobych was held during the work of the seminar with participation of the leading specialists in the field of automation of scientific research and personnel training.

The participants of the seminar recommend that the next school-seminar be held in 1983.

6521
CSO: 1863/72

USE OF COMPUTER EQUIPMENT TO CONTROL PRODUCTION PROCESSES IN INSTRUMENT BUILDING

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 110-111

[Article by Candidate of Technical Sciences Igor' Semenovich Yeremeyev, Kiev, and methodologist Galina Alekseyevna Pimenova, Sevastopol Branch of RDENTP, Sevastopol]

[Text] The republic scientific and technical conference "Use of Computer Equipment to Control Production Processes in Instrument Building in Light of the Decisions of the 26th CPSU Congress," organized by the Sevastopol Branch of RDENTP [not further identified] of the Ukrainian SSR Znaniye Society, the Voroshilovgrad Oblast Administration of the NTO [Scientific and technical department] Priborprom imeni S. I. Vavilov and the Sevastopol Instrument Building Plant imeni V. I. Lenin, was held in Sevastopol from 17 through 20 August 1982.

More than 100 specialists from 30 of the country's cities participated in the work of the conference.

The following problems were considered in the reports and communications (more than 60) included in the work schedule of the conference and also in unscheduled communications:

the use of computer equipment in monitoring and diagnostic systems;

the use of computer equipment in ASUTP [automated production process control system];

software for monitoring and diagnostic systems and ASUTP.

Among the reports and communications of the first group that evoked the greatest interest of the conference participants, one can include the following: V. P. Sidorenko "Use of Computer Equipment for Automation of Production Processes by UVK [control computer complexes]," devoted to description of the automated multiterminal monitoring and diagnostic complex KODIAK, developed at the PO [production association] Elektronmash (Kiev), on the basis of the SM-4 central control computer and a number of problem-oriented devices (up to 16),

operating in the time-sharing mode and that permit monitoring of logic and analog units; that of V. S. Klevchuk "The Use of Computer Equipment to Control Production Processes at the PO Elektronmash," in which a system for controlling the process of monitoring power supply sources by the SM EVM [International small computer system] during acceptance-delivery trials is described that provides organization of the sequence of performing monitoring operations for several types of power supply sources in the time-sharing mode, recording of deviations of parameters from the norm, display of the status of the system at the user's request, monitoring of installation and other procedures; that of R. Ye. Boguslavskiy et al "Apparatus for Monitoring and Diagnosis of Digital Circuits by the Signature-Probability Method," devoted to description of the Versiya-128 monitoring and diagnostic system, developed at the SKB [special design office] of the PO Rele i avtomatika (Kiev) and that represents a signature-probability bench capability of operating both in the autonomous mode and as a system terminal based on the Elektronika-60 computer (connection of up to 24 benches is possible in the latter case).

A number of reports and communications of the first group were devoted to problems of monitoring and diagnosis of MPK [not further identified] and BIS [large integrated circuit] and also monitoring of installation.

Among the reports and communications of the second group, the following reports evoked considerable interest: that of N. P. Belikikh and S. A. Maslyuk "A Hardware Complex for Automated Monitoring and Control Systems," that of G. M. Lekhnova "Development of the SM-2 for Use at the Upper Level of Hierarchical ASU [automated control system]," that of V. V. Rezanov and V. M. Kostelyanksiy "The Status and Prospects for Development of Computer Equipment for Automated Production Process Control Systems," and also that of V. Ya. Rozentsvayg "SM EVM and ASVT-PS [not further identified] for ASUTP." The main characteristics of the hardware complex of the second-generation SM-2 family, created by the NPO [scientific production association] Impul's (Severodonetsk), the PS-3000 multiprocessor complex and also the problem-oriented microprogrammable automatons based on the SM-1634 processor were outlined in these reports.

The experience of the PO Elektronmash (Kiev) in development of graphic displays, line printers (including laser printers with xerography and output of data on a large screen), flatbed plotters and other equipment for SAPR [automated design system] based on the hardware complex of the SM-3 and SM-4 computers was generalized in the report of A. Ya. Budyanskiy "Experience and Prospects for Development of Peripheral Equipment for UVK [control computer complex]."

Reports and communications devoted to automation of installation work in instrument building and to the problem of using microprocessor complexes and microcomputers in ASUTP were also represented in this group.

The reports of V. A. Nedonosenko et al "The Problem of Software of Automated Installation Monitoring Devices," of A. N. Zakharov et al "Programs of the Functional Monitoring Complex of Logic Units," the communications of A. Ya. Budyanskiy "The Capability and Feasibility of Using Methods of Experimental

Mathematical Theory for Monitoring and Diagnosis of Logic Units" and "The Use of Computers for Monitoring and Diagnosis of Logic Units by MTE [magnetic triode component] Methods," and also of I. S. Yeremeyev "One Method of Increasing the Failure Stability of Process Control Systems" evoked definite interest among the third group of reports.

The discussion that was organized during debate of the reports and communications touched on problems of increasing the efficiency of using computer equipment in production process and monitoring and diagnosis control systems in instrument building, more extensive use of new-generation mini- and microcomputers and also promising peripheral devices.

The work of the conference contributed to exchange of advanced experience in development and introduction of ASUTP, automated monitoring and diagnostic systems and also to propaganda of modern firmware for automation of production processes and leading methods of using them in instrument building technology.

6521
CSO: 1863/72

IMPROVING MANAGEMENT OF PRODUCTION ASSOCIATION BASED ON STANDARDIZATION AND COMPUTER EQUIPMENT

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 112-114

[Article by engineer Petr Yakovlevich Kalita, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] The republic seminar-conference "Improving the Management of the Production Association on the Basis of Standardization and Computer Equipment," organized by the Scientific and Technical Council for Work Quality and Efficiency attached to the Kiev Gorkom of the Ukrainian communist party, the scientific council on the problem "Cybernetics" of the Ukrainian SSR Academy of Sciences, the Institute of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences, by the All-Union Scientific Research Institute of Standardization and by the Kiev city Soviet of the NTO [Scientific and technical society], was held in Kiev from 21 through 23 September 1982.

A total of 150 persons--managers and specialists of the leading production associations and industrial enterprises of the Ukraine and Belorussia, Moscow and Leningrad and scientists of the leading institutes of Gosstandart [State Committee for Standards], Ukrainian SSR Academy of Sciences, Ukrainian SSR Minvuz [Ministry of Higher and Secondary Specialized Education] and sector scientific research institutes--participated in the work of the republic seminar-conference.

The deputy chairman of the NTS [Scientific and technical council] on work quality and efficiency attached to the Kiev Gorkom of the communist party of the Ukraine Candidate of Economic Sciences G. I. Kalitich opened the conference. He informed those gathered of the existing Kiev municipal system for control of work quality and efficiency, approved by Gosstandart and recommended for extensive dissemination in other regions of the country, and emphasized that a special place is being allocated in this system to its most numerous sections--production associations and industrial enterprises. To further improve the management of these sections, it is very effective to achieve fundamental combination of modern methods and means of standardization and automation of control processes in the planned control systems.

The prospects and basic directions for improving the management of production associations and industrial enterprises on the basis of standardization in

light of the decisions of the 26th CPSU Congress were considered in the report of Candidate of Economic Sciences I. I. Chayki (VNIIS [All-Union Scientific Research Institute of Standardization]). A great deal of attention was devoted to the methodical bases for development and introduction of association and enterprise management systems based on the complex GOST 24525-80 "Management of the Production Association and Industrial Enterprise." The reporter dwelt on individual problems of practical organization of experimental introduction of standards, positively characterized work on introduction of the complex of GOST carried out by the city NTS for work quality and efficiency in the capital of the Ukraine and noted the positive experience of cooperation of VNIIS with organizations and enterprises of the Ukraine in development and improvement of the regulations of the complex of GOST 24525-80, including cooperation with the Institute of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences.

In his report, Doctor of Physicomathematical Sciences V. V. Shkurba (Institute of Cybernetics, Ukrainian SSR Academy of Sciences) considered the trends for development of ASU during the 11th Five-Year Plan and modern approaches to design of enterprise management systems based on computer equipment. He noted that modern and especially promising ASU are being constructed as automated workstation systems which become the main systems-forming modules of the ASU. New simulation functions such as instrumentation of planning, design, economic analysis and accounting are being developed and formulated organizationally in management of the national economy and its individual sections. ASU informatics is being structured in a new way--data display, denomination and management of information processes acquires an integrated "production" nature. These features should find the corresponding reflection in the GOST complex and specific production association management systems based on enterprise standards and modern computer equipment. Therefore, such institutes as VNIIS and the Institute of Cybernetics, Ukrainian SSR Academy of Sciences, and also other organizations and enterprises that are leaders in various aspects of management of social production must cooperate.

In his report, Doctor of Technical Sciences L. G. Shatikhin (KTILP [Kiev Technological Institute of Light Industry]) considered the essence of the systems approach and systems development of the structure of entities on different examples; it was suggested that the widely accepted method of structural matrices be used for analysis and synthesis of complex systems. The essence of this method is that the system to be investigated is displayed in the form of block matrices, sequentially detailed upon transition from a higher hierarchical level to a lower level; in this case the relationships between the individual elements contained in it are followed on the matrix itself.

In his report, P. Ya. Kalita (Institute of Cybernetics, Ukrainian SSR Academy of Sciences) noted that transition from the level of structural-functional development to engineering design of the entire management art becomes important at the modern phase. Along with development of production processes of functional management, greater attention must be devoted to design of production processes of holistic (interfunctional) management, i.e., processes of data analysis and formulation of decisions oriented toward management of specific "object" parameters of a production system having a complex hierarchical

structure. The methodical approach to design of object management technology using modern informatics and computer equipment was proposed and problems of practical realization of this approach in real management systems, including those developed according to the requirements of the complexx GOST 24525-80, were considered.

In her report, Candidate of Physicomathematical Sciences T. P. Podchasova (Institute of Cybernetics, Ukrainian SSR Academy of Sciences) formulated the basic problems of placing production management on a mathematical basis under conditions of development and functioning of ASU and considered the possible methods of solving them. The various approaches to solution of industrial enterprise management problems were considered in detail from the aspects of systems optimization and the suggested approaches were illustrated with examples of results of practical introduction of them.

The speech of Candidate of Economic Sciences K. F. Yefetova (Institute of Cybernetics, Ukrainian SSR Academy of Sciences) was devoted to problems of formulation of the production product output program for several criteria. Models of formulation of the norms of laboriousness and tasks to reduce them were considered with bringing them to the corresponding subdivisions of industrial enterprises.

In his report, Candidate of Economic Sciences Ye. B. Bibik (Institute of Cybernetics, Ukrainian SSR Academy of Sciences) considered the method of designing the organizational structures of management systems, developed at the Institute of Cybernetics, Ukrainian SSR Academy of Sciences; this method can be used both to improve management of existing production and to design organizational support of the management systems of associations and enterprises under construction.

In his report, Candidate of Technical Sciences M. A. Goriyev (NPO [scientific production association] Gorsistemotekhnika) considered an automated interactive operational decision-making system--DISPOR--and problems related to the use of the system in different types of plants.

Candidate of Technical Sciences V. M. Yurkov (NPO Gorsistemotekhnika) acquainted the conference participants with the basic principles and features of design and functioning of an automated design system for ASU. He also considered problems of increasing the efficiency of production association management systems, designed by using SAPR ASU [automated design system for ASU].

In his report, A. A. Rakov (VNIIS) outlined in detail the problems of the method and practice of designing a specific resource management subsystem according to the requirements of GOST 24525.5-81.

Problems of certification of supplier enterprises on the quality of products delivered by them and improvement of management of fulfilling the production and product delivery plan were outlined in the report of V. G. Guba (VNIIS).

The conference participants became acquainted with great interest with the practical experience of improving management at a number of leading production

associations and industrial enterprises of the Ukraine, including those at the Sumy Machine Building Production Association imenie Frunze (the reporter was Candidate of Economic Sciences M. F. Balan), the Kiev Production Association for Knitted Fabrics imeni R. Luxemburg (reporter was G. I. Rudenko), the Kiev Experimental-Demonstration Reductor Plan (the reporter was V. D. Suprunenko) and others.

Having discussed the presented methodical materials, scientific and practical experience and also the existing problems in improvement of production association management, the conference participants adopted developed recommendations in which it was noted that the most important direction for further development and improvement of production association management systems based on standardization and computer equipment should be development and introduction of integral management technology. It was recommended that production associations and industrial enterprises:

become more active in work to develop and further improve management systems based on the systems and complex approach using methods and means of standardization, modern informatics and computer equipment and with regard to all the leading experience accumulated in this field;

participate actively in improvement of the complex of state standards for management of the production association and industrial enterprise, analyze its positions and send substantiated and constructive proposals to finish them to VNIIS.

extensively use the methodical approach of engineering design of integral management technology, suggested by the Institute of Cybernetics, Ukrainian SSR Academy of Sciences, including the technique of data analysis and decision-making on deviations of the parameters of control entities, in improving management systems;

put into wide practice the extensive structural and logical analysis of control entities at the stage of the predesign survey that permit one to determine and display all the significant elements of control entities and the relationships existing between them and so on.

The republic seminar-conference appealed to the State Committee for Science and Technology, State Committee of Standards and the USSR Academy of Sciences with a request to consider the problem of combining and coordinating the efforts of all organizations involved in working out problems of management using various methods (standardization, automation, economic, legal and so on) to guarantee a systems and complex approach to solution of the problems of improving the management of production associations and industrial enterprises.

The conference participants felt it was feasible to hold the All-Union Scientific and Practical Conference "Improvement of the Production Association Management Technology on the Basis of Standardization and Computer Equipment" in Kiev during the third and fourth quarter of 1983.

6521
CSO: 1863/72

PROBLEMS OF DEVELOPING PROGRAMS FOR CALCULATING PROCESS CONTROL

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 114-115

[Article by Candidate of Technical Sciences Boris Nikolayevich Pan'shin, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] The seminar "Problems of Developing Programs for Calculating Process Control," organized by the Republic House of Economic and Scientific and Technical Propaganda, the Institute of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences and the machine building pavilion of the Ukrainian SSR VDNKh [exhibition of achievements of the national economy], was held in Kiev from 21 through 24 September 1982. The seminar has become traditional--it has been held in Kiev at the end of September since 1978 and approximately one-third of the visitors are its constant and most active participants.

The main task of the seminar is to organize exchange of experience in development and operation of the programs for control of the calculating process at departmental VTs [computer center] and VTsKP [multiple-user computer center] and in computer networks. The timeliness of the seminar topic was again confirmed this time by participation of specialists from the RSFSR, the Ukraine, Belorussia, Moldavia, Estonia and other fraternal republics in its work. More than 150 specialists from more than 135 of the country's different organizations, mainly employees of computer centers, participated in the work of the seminar.

A total of 46 reports and communications, which could be conditionally divided into the following groups, was heard and discussed at the seminar:

development and operation of programs for automation of organizational and economic process control functions at modern computer centers (VTsKP);

development and operation of programs for production process management of solving applied problems;

improvement of existing and development of new organizational flow charts for exploitation of the resources of computer centers (VTsKP).

The following among the reports and communications of the first group evoked the greatest interest: "The Orakul-2 Interactive System for Organization of the Calculating Process at Multiple-User Computer Centers" (V. A. Kushnirov and F. A. Levchenko--SKTB [special design and technological office] of the PO [production association], Institute of Cybernetics, Ukrainian SSR Academy of Sciences), "Operating Experience of Applied Program Packs to Manage the Data Archives of VTsKP" (G. S. Serykov and A. Ye. Selyuchenko--SKTB PO, Institute of Cybernetics, Ukrainian SSR Academy of Sciences), "Experience of Development and Operation of Diagnostic Data Processing Programs on Breakdowns of YeS EVM [Unified Computer System] devices" (L. K. Snigireva--Institute of Cybernetics, Ukrainian SSR Academy of Sciences) and "Development and Operation of the Program Complex Administrative System of VTsKP" (A. F. Tutov--Institute of Cybernetics, Ukrainian SSR Academy of Sciences).

Reports devoted to development and operation of devices and methods for analyzing the efficiency of using computer center resources evoked great interest among the seminar listeners. It was noted that a number of developments have appeared during the past 3 years, the practical application of which has made it possible to determine and formulate more clearly the basic requirements on the indices of the operating efficiency of computer centers and on data preparation programs for calculation and analysis of these indices. These are primarily those methods and complexes of programs such as:

ORGVYTs-1 and ORGYTs-2 complexes accompanied by the NPO [scientific production association] Tsentrprogrammsistem (Kalinin);

the automation of resource accounting complex, developed at the computer center of the PKB [planning and design office] of management systems, Estonian SSR Minlegprom [Ministry of Light Industry] (Tallinn);

the Protsess complex, developed at the computer center of the Latvian SSR UGA [office of the city architect] (Riga);

the KIBORG complex, developed at the computer center of the Institute of Cybernetics, Ukrainian SSR Academy of Sciences.

The following reports were devoted to these problems: "approaches in determination of the economic analyses of the level of organization of the calculating process at VTsKP" (A. R. Myagi--computer center of planning and design office of statistical administration, Estonian SSR Minlegprom, Tallinn), "Supplementation of the Accounting-Monitoring Program Complex of Means of Accounting for Start-Up of User Procedures" (V. P. Platonov and G. A. Zvereva--VNII [All-Union Scientific Research Institute] of Hydrometeorological Data, Obninsk) and "Management of the Calculating Process in Operation of Large Program Systems" (S. A. Kozlov and D. A. Fateyev, NITsEVT [scientific research center for computer equipment], Moscow).

The report "Improving the Resource Accounting Automation Systems of Multiple-User Computer Centers" (N. N. Gudim, Moscow) was a unique summary of the discussion organized on the problem of accounting programs.

The opinion of seminar participants on the need for the most rapid completion of work to standardize indices to evaluate the operating efficiency of a modern computer center and development of a universal accounting-measuring system delivered together with the operating system based on these standards was the general conclusion on the problem of organization of accounting for the use of information computer resources.

Among the data formulated by this standard system should be included those that permit calculation of both the quantitative and qualitative indices of the efficiency of organizing the calculating process.

Among the reports of the second group, devoted to development of programs for management of the technology of solving applied problems, the following evoked the greatest interest: "Use of a Real-Time Supervisor to Design Calculating Process Management Systems in the OS YeS [operating system of unified computer system]" (A. I. Novokhatskaya and A. I. Pleshch--Institute of Cybernetics, Ukrainian SSR Academy of Sciences), "Program Complex for Management of Problem-Solving Technology in Complex VS [computer network]" (A. G. Stepanenko, V. S. Kirilyuk and Ye. V. Andreyeva--Institute of Cybernetics, Ukrainian SSR Academy of Sciences), "Planning the Solution of Regular Problems in ASU Developed on the Basis of Computer Networks" (E. G. Petrov and Ye. A. Tyricheva--KhIRE [not further identified], Kharkov), "The Configuration and Design Principles of Programs for Organization of the Calculating Process in ASU" (A. Ya. Shafrov--GPKI [State Planning and Design Institute] of ASU, Ivanova), "Optimization of the Parameters of the YeS Operating System Based on Hybrid Simulation Data" (A. I. Vinnichenko and V. G. Litvin, Moscow), "Development and Operation of Specialized Operating Systems" (Yu. I. Mal'tsov--NII AEM [not further identified], Tomsk) and "Experience of Operating the Programs of a Two-Machine Complex of BESM-6 Computers at the Multiple-User Computer Center of the Institute of Cybernetics, Ukrainian SSR Academy of Sciences" (Zh. A. Kalenchuk-Parkhanova--Institute of Cybernetics, Ukrainian SSR Academy of Sciences).

The seminar participants discussed with great interest the reports of the third group, devoted to improvement of the organizational resource operation flow charts of multiple-user computer centers, efficient use of organizational equipment at computer centers and making more efficient the user-computer center resources relationships. These reports mainly generalized the experience of organizing the calculating process at the multiple-user computer center, Institute of Cybernetics, Ukrainian SSR Academy of Sciences.

The following evoked the greatest interest among this group of reports: "Automation of Data Processing Monitoring Functions and Use of Magnetic Carrier Archives" (V. N. Nazarov and V. A. Shatalov), "System for Nonperforation Data Preparation Based on the SPD-9000 and Devices of the YeS Operating System" (V. S. Laznyuk) and "Experience of Organizing the Brigade Contract in the Operator Service of the Multiple-User Computer Center, Institute of Cybernetics, Ukrainian SSR Academy of Sciences" (V. A. Shatalov, A. M. Ivanov and A. I. Sharovara).

Survey reports on development of operating systems and programs for management of the calculating process, given by A. L. Alayev and L. B. Gradus (GIAP [State Scientific Research and Planning Institute of the Nitrogen Industry and Products of Organic Synthesis], Moscow), the communication of the new KORKORD remote program debugging system, made by G. M. Grushanskiy (VNIIPI ASU [All-Union Scientific Research and Planning Institute for Automated Control Systems] of the Light Industry, Ivanova), the communication about a modified method of organizing mutual calculations with users of the multiple-use computer center, made by P. Ts. Drukhoy (Multiple-user computer system, USSR Minenergo [Ministry of Power and Electrification], Tashkent), the communication on automation of functions of the external control circuit of the multiple-user computer center, given by S. N. Fokin (Minsk), the communication on the experience of operating multiple-user systems, made by M. I. Shlosberg (Giprotranssignalsvyaz' [State Planning and Surveying Institute for the Planning of Signalization, Centralization, Communications and Radio in Railroad Transportation], Leningrad) and so on were heard with great attention and interest at the final session of the seminar.

The exhibition of prospectuses of new developments of devices for management of the calculating process and excusions to computer centers, organized at the Science Pavilion of the Ukrainian SSR VDNKh and at the multiple-user computer center, Institute of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences, organized by the reporters, contributed to the fruitful work of the seminar.

The participants viewed a film about V. M. Glushkov at the final session of the seminar.

The results of the work of the seminar were summarized in a number of recommendations, in which special attention was devoted to the need to consider modern computer centers (in solution of any problems--engineering, production, personnel and so on) as unique "industrial data processing factories" that comprise the basis for a machine informatics sector established in our country, the cornerstone concepts of formation and establishment of which were formulated in the last works and speeches of Academician V. M. Glushkov.

The seminar participants recommended that the work of the seminar be continued and that the next session be held in September of 1983 at Kiev.

6521
CSO: 1863/72

PROBLEMS OF THEORY AND DESIGN OF DATA CONVERTERS

Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 6, Nov-Dec 82 pp 115-116

[Article by Candidate of Technical Sciences Petr Stepanovich Klochan, Institute of Cybernetics, Ukrainian SSR Academy of Sciences, Kiev]

[Text] The republic conference "Problems of Theory and Design of Data Converters," organized by the Institute of Cybernetics imeni V. M. Glushkov, Ukrainian SSR Academy of Sciences, the Republic House of Economic and Scientific and Technical Propaganda of the Znaniye Society, Ukrainian SSR, and also jointly by the Vinnitsa and Odessa Polytechnical Institutes and the Kharkov Institute of Electronics imeni Academician M. K. Yangel', was held at Vinnitsa from 7 through 9 September 1982. The Ukrainian Administration of NTO [scientific and technical department] Priborprom and the following Vinnitsa organizations--the Oblast Znaniye Society of the Ukrainian SSR, the House of Technology of NTO, the PO [production association] Oktyabr' and the TsKB [central design office] of Information Technology--participated actively in preparation and conducting of the conference.

A total of 270 specialists representing 62 organizations from 35 of the country's cities participated in the work of the conference. A total of nine plenary reports and 86 reports and communications at meetings of three sections was heard and discussed.

In opening the conference, the chairman of the organizing committee A. I. Kondalev noted the basic achievements in the field of Soviet converter equipment, considered a number of the most important trends in solution of the problem of data coding, approximation and compression at PFI [data format converter] and pointed out the need for further development of high-performance converter equipment devices and industrial assimilation of them in the form of SIS [medium-scale integrated circuit] and BIS [large-scale integrated circuit].

In his speech, the Prorector of the Vinnitsa Polytechnical Institute V. S. Osadchuk acquainted the conference participants with the promising developments of the institute in the field of converter equipment that merit widespread introduction into the country's national economy.

Reports were heard at the plenary sessions that reflect the basic directions in development of converter equipment. Specifically, classification of analog-digital microprocessors was suggested in the report "Analog-Digital Microprocessors for Gathering, Preliminary Processing and Recording of Analog Signals" (V. B. Smolov, Ye. P. Ugryumov and I. V. Gerasimov), their systems functions and characteristics of software and programs were considered and comparative analysis of the different configurations of analog-digital microprocessors was given.

Problems of design of high-precision PFI, which utilize Fibonacci and the "gold" proportion codes, were considered in the report "Prospects for Development of Data Format Converters Based on Codes with Irrational Bases" (A. P. Stakhov and A. D. Azarov). The advantages of these devices were pointed out when they are used in digital audio recording and systems for automation of acoustic and medical research.

The results of developing an incremental ATsP [analog-digital converter] for the high-performance macroconveyer computer developed at the Institute of Cybernetics, Ukrainian SSR Academy of Sciences, comprised the basis of the report "Problems of Increasing the Productivity of PFI in Problem-Oriented Computer Systems" (Yu. A. Brayko).

Problems of development, circuit engineering and basic parameters of the high-speed ATsP, prepared for serial production at the Kiev PO [production association] Tochelektropribor, were considered in the report "High-Speed Analog-Digital Display" (V. A. Bagatskiy, V. P. Mel'nichenko and N. N. Mironets).

The report "Biomedical Data Input Device in the SM-1800 Microcomputer (V. F. Kirven', I. A. Averin and V. N. Mamayev) was devoted to practical development.

The report "Technical Realization of Geometric Progressions as Bases of a Calculating, Measurement and Calculating Device" (N. A. Filipov) in which the features and capability of using multiplicative calculating systems in data gathering and processing systems, was also heard at the plenary session.

Problems of simulation and computer-aided design, to which the following reports were devoted: "Joint Simulation of Firmware of ATsP of Data Gathering Systems" (I. V. Gerasimov), "Simulation of Processes of the Occurrence and Distribution of Errors in Linear Dimension and Displacement Meters at PPZ [not further identified]" (N. V. Bessarabov and Ye. S. Sayenko), "Design of SAPR [automated design system] SI [not further identified] Based on the International Small Computer system" (A. V. Makarov), "Problems of Computer-Aided Design and Optimization of Functional ATsP for Monitoring the Characteristics of Random Processes" (N. N. Atamanenko) and so on--were mainly considered and discussed in the first section.

The following basic directions can be distinguished in the work of the second section:

high-performance PFI, their subassemblies and components on a modern microcomponent base. For example, the reports "Survey of Methods of Design

and Circuit Engineering of ATsP and TsAP [digital-analog display] of Increased Accuracy and Speed" (V. P. Martsenyuk and V. I. Moiseyev), "Analog-Digital Converter of Increased Accuracy" (V. A. Romanov), "ATsP of Increased Speed for Strain Measurement" (G. V. Sotov, V. V. Skalevoy and M. V. Skalevoy), "Multithreshold Comparator for Parallel-Series ATsP" (V. P. Stokay) and so on;

analysis of the modern characteristics of microelectronic ATsP and TsAP and methods of improving them: the reports "Analysis of the Precision of Tracking Frequency-Pulse Data Converters" (V. F. Mitin), "Analysis of the Metrological Characteristics of Random Signal Converters" (V. A. Fabrichev), "Analog Converters of Voltage Difference to Deviation of Pulsed Signal Parameters" (V. I. Koval'kov, V. A. Azarkin and A. A. Bakhtiozin), "Integrated ATsP for Multichannel Sensor Monitoring System with Low Signal Levels" (A. P. Konopkina and R. V. Kurdydyk) and so on;

monitoring and measurement of converter parametes: the reports "Experimental Investigation of ATsP Errors in the Dynamic Mode During Random Signal Conversion" (V. I. Konovalov, A. A. Bakirov and G. A. Zil'berman), "Measurement of Coding Noise in Multidigit ATsP and TsAP of Wideband Communications Signals" (S. S. Kogan and A. G. Likiardopulo) and so on;

The reports of the third section were mainly devoted to:

methods of design of converter information on the basis of redundant codes, for example, the reports "Use of Redundant L-Codes in Data Conversion Systems" (N. V. Alipov), "Possibilities of Developing ATsP and TsAP in Codes with Irrational Bases on the Basis of Hybrid Microcircuits" (A. D. Azarov, I. G. Gavrilyuk, V. P. Volkov) and so on;

the use of PFI in automation systems, for example, "Method of Increasing the Systems Productivity of PFI" (V. N. Lavrent'yev), "A Method of Data Processing in Automation of Audiometric Research" (I. P. Tynskiy), "Integration of Microelectronic ATsP with the Multiplex Channel of an Information Signal" (S. N. Kuznetsov and Ye. M. Tverskoy) and so on.

As a result of discussing the reports and discussions, the conference participants noted the high level of individual developments which can be effectively utilized in the country's national economy. Considerable success was achieved in development of analog microprocessors, in systems use of PFI, in development of methods of converter design based on redundant codes and in developments of ATsP for strain measurement. Industrial production of 10- and 12-digit ATsP and TsAP in the form of SIS and BIS was organized. Some developments are not inferior to the best worldwide specimens. Most developments are carried out within the framework of complex and specific union and republic programs.

At the same time, the conference participants are turning the attention of organizations and specialists in the field of converter equipment to the timeliness of solving the following problems:

development of applied and theoretical research in the field of analog-digital and digital-analog conversion;

development of the fundamentals of computer-aided design of data converters that meet systems requirements on engineering, metrological, operating and economic characteristics in their indices;

development and industrial assimilation of microelectronic converter equipment devices (specifically PFI with built-in microprocessors) using new physical processes and phenomena, redundant codes, noise-suppressing structures, functional converters with approximation of orthogonal functions and also PFI and their components in the form of SIS and BIS:

development and serial production of systems PFI using microprocessors for digital television and communications, sound recording, scientific research and control systems;

introduction of methods and equipment for automation of production processes and quality control of components, subassemblies and devices of converter equipment.

The conference participants plan to hold the Fifth All-Union Symposium "Problems of Developing Data Format Converters" in 1984 at Kiev.

6521
CSO: 1863/72

PUBLICATIONS

TABLE OF CONTENTS OF JOURNAL 'AUTOMATIC MONITORING AND MEASURING', JANUARY-FEBRUARY 1983

Novosibirsk AVTOMETRIYA in Russian No 1, Jan-Feb 83 p 1

[Text] Table of Contents

9875
CSO: 1863/99

- END -