JPRS L/10017

28 September 1981

USSR Report

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 22/81)

FBIS FOREIGN BROADCAST INFORMATION SERVICE

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

NOTE

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

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

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

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

JPRS L/10017

28 September 1981

USSR REPORT

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 22/81)

CONTENTS

HARDWARE

PUBLICATIONS

HARDWARE

BUILDING VIDEO TERMINALS FOR ON-LINE INTERACTION

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 5, May 81 pp 7-8

[Article by V. A. Stryuchenko: "Characteristic Features of the Building of Video Terminals for On-Line Interaction"]

[Text] Video terminals providing remote man-machine interaction permit on-line exchange of different types of information in teleprocessing systems. This interaction is in the nature of an informational conversation since it involves the operations of receiving, storing, retrieving, displaying, processing and transmitting information.

Systems analysis of the structural organization of remote video terminal equipment and of their operating algorithms with respect to functional characteristics has made it possible to distinguish the inter-related functional parts in the structure of this equipment as broken down into data display, data input and preparation (editing) and data exchange (Figure 1). Simultaneous operation of these inter-related control circuts with the participation of the human operator characterizes the functioning of video terminal equipment.

In the stage of video-terminal design this structural division facilitates the development of requirements for individual units and assemblies and the algorithms involved in the functioning of video terminals and permits classification of video terminal hardware.

Data display - this is the process of converting information from digital form into screen displays or the documentation of data for presentation to the operator in visual form and then the processing of this data. Data is presented in the form of either text or graphic displays. Of great importance in this regard are the principle of image formation, image formats, the methods employed to extract individual fragments and image quality. After visual perception of the information generated and making his decisions the operator inputs and prepares data by means of a variety of input devices: keyboard, light pen etc. The speed of this information-processing channel, that is, the possibility offered of on-line data editing and input, to a great extent determines the capacity of the entire interactive subsystem. The data exchange process includes the conversion of information for transmission via communication links, execution of exchange algorithms and conversational procedures and is governed by the rate at which data is received and transmitted and by the interaction of the video terminal with other teleprocessing system hardware.

Some of the control processes in video terminal systems are cyclical in nature and occur without operator participation, the regeneration of a display on screen, for example, or mode synchronization, computer-initiated message output or diagnostic checks.

Video terminals may be categorized with respect to their functional characteristics as follows:

image - by type of display (screen), method of image formation and form of information presentation;

data input and preparation - by degree of sophistication of input and editing functions and by control procedure;

data exchange - by type of conversation, exchange procedure and rate of data transmission.

Efficient on-line interaction in conversational systems requires attention to the improvement of the functional characteristics of video terminal equipment. For one thing, video terminals with CRT, now most widely used, do not provide best-quality screen images, the possibility of improving which is governed by the need to improve a number of parameters: to reduce raster distortion and image instability (static and dynamic) due to external factors, improve the accuracy of the addressing of image elements, brightness, clarity (focusing) and to eliminate interframe flicker. Degradation of these parameters may shorten the period of time an operator can work with the screen and result in a sharp drop in operator productivity.

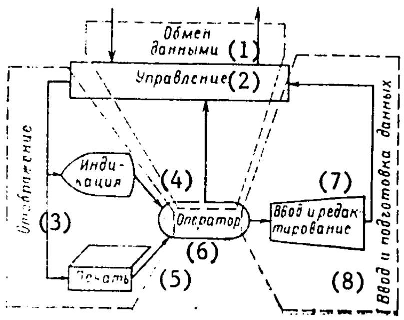


Figure 1. 1 - data exchange; 2 - control; 3 - image; 4 - - display; 5 - print; 6 - operator; 7 - input and editing; 8 - data input and preparation.

The need to regenerate images on a CRT screen and establish the required frame frequency to insure a stable image dictate the use of complex control algorithms to coordinate the parallel operation of display, i/o and editing channels, which creates equipment redundancy, the use, for example, of multilevel storage in the system and of high-speed parallel algorithms in control. Figure 2, for example, shows how the operational cycle of a video terminal is complicated if it insures image stability in all modes (Figure 2b) as compared with the realization of successive cycles of video-terminal operation with interruption of the CRT screen display during the time required for data exchange (Figure 2a).

The above-mentioned are still to a large degree characteristic of character-graphic video terminals, with the development of which are also associated problems with providing output to the screen of mixed information with the use of raster and vector (or coordinate) imaging methods for editing graphic information. These problems are connected primarily with the coordination of the time characteristics of image regeneration and other operational modes.

One of the promising directions toward solution of these problems is the creation of a video-terminal display channel using gas-discharge (plasma) displays (GDD). A matrix imaging system, which provides good accuracy, stability and illumination characteristics along with adequate resolution capability, permits generation of high-quality character-graphic images on plane plasma display screens. The table below presents comparative characteristics of GDD and CRT. The use of ac GDD in video terminals permits establishment of combined information transmission and editing modes (exploiting the GDD's ability to store information without regeneration), simplification of control and display algorithms and lower speed requirements on output units synthesizing images on the screen. This in turn permits reduction of memory capacity to that required for image editing. The small size and weight of the GDD permits convenient installation at the operator's work position.

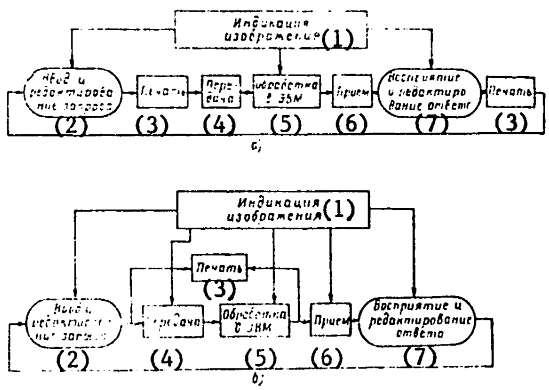


Figure 2. 1 - image display; 2 - query input and editing; 3 - print; 4 - transmission; 5 - computer processing; 6 - reception; 7 - reception and editing of response.

Video terminals with plasma screens have virtually no analog components; all control functions, including display, are executed by digital components. This permits inceased accuracy and stability of system parameters. Fabrication of video terminals with plasma displays at the same time requires taking into account a number of special characteristics connected with the use of GDD. For one thing, because of the relatively low writing speed (20 μs) of the element, the use of GDD requires that steps be taken to increase the speed with which information is output to the screen, which is in fact achieved by providing element group writing on a panel by means of informational analysis of output data with the use of two element writing speeds and image scanning. It is also necessary to insure the reliable firing (writing) of individual image elements over the entire operating field, by introduction, for example, of preparatory writing modes. Information has also to be specially organized and edited on a screen providing selective erasing and restoration of graphic image elements.

Table. AC GDD and CRT: Comparative Characteristics.

			CRT	
Characteristic	AC GDD		Kinescopes	Special
Maximum resolution in number of points	512 X 512 (1024X1024)		600 X 600 (1000X1000)	2000 X 2000
Minimum point size in mm		0.4-0.5		0.2-0.3
Maximum size in mm	300 X 300		400 X 400	250 X 250
Maximum brightness in cd	100 - 200		100-150	
Color of illumination	orange (green)		white	white (green)
Number of brightness gradations	1		8	-
Geometric distortions in %	<0.5		2-5	2-3
Nonlinear distortions in %	<0.5		5-10	5
Image dimensional instability in %	-		3-5	
Scanning angle in ang. deg.	>120		<90	
Time to write (erase) any point on screen in μs	20		0.1-50	-
Image storage without regeneration	Yes		No	limited in storage CRT

Characteristic	AC GDD	CRT	
		Kinescopes	Special
Selective erase/write (edit) capability	Yes		limited in storage tubes
Screen-control principle	discrete	analog	
Maximum supply voltage in V	250	18,000	20,000
Permissible supply voltage fluctuation in %	±2	±5; ±10	
Maximum display depth dimensions in mm	10-60	200-400	>500
Weight in kg	1.5-3	10-15	

8963
CSO: 1863/202

OPTICAL-FIBER PRESSURE TRANSDUCER

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 5, May 81 pp 21-22

[Article by V. I. Busurin, candidate of technical sciences, V. I. Pasynkov, engineer, and N. P. Udalov, doctor of technical sciences: "Optical-Fiber Pressure Transducer"]

[Text] Recent advances in fiber-optics technology have made possible the successful application of fiber light guides in the development of a number of optical and opto-electronic instruments, including optical-fiber pressure transducers--OFPT [1-3].

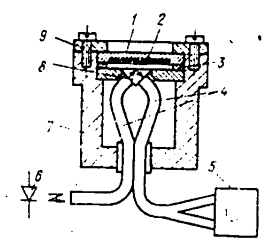
Interest in OFPT may be explained by characteristics distinguishing them to advantage from other pressure transducers: insensitivity to electromagnetic fields, high speed, small size and low weight. Their high heat resistance (to the point of physical destruction) and thermal stability are the result of the absence of thermally unstable components in the transducer's sensitive element, since with the use of light guides the transducer's entire semiconductor component can be moved as far as desired away from the pressure measuring area. Thermal instability due to changes in the modulus of elasticity of the membrane may be virtually eliminated in this instance with the use of a quartz membrane [4]. The absence of current-carrying components in these transducers makes the latter indispensable in situations not permitting the presence of current-carrying parts, when the immediate environment is dangerously explosive, for example, or the medium involved is under high electric potential.

Fabrication of OFPT conventionally employs either deformation of the reflecting surface of the membrane deflecting under the effect of pressure [1, 2] or redistribution of the light flux reflected from a moving reflector attached to the membrane [3]. The linearity and sensitivity of these transducers are determined by the geometric parameters of the housing and the light guides, the distance between membrane and light guide, etc. To obtain characteristic linearity, for example, it is necessary to increase this distance, which in turn increases optic losses [1].

The OFPT illustrated in Figure 1 is based upon the principle of the destruction of total internal reflection (TIR) [5] and functions in the following manner. The membrane deflects with pressure and varies the gap between the light-absorbing layer and the base of the prism, which causes partial destruction of TIR on the upper face of the prism. Part of the radiation passes through the upper face and is absorbed in the light-absorbing layer. The change in the gap thus produces a change in the intensity of the reflected light striking the photosensitive bridge circuit.

For the steady-state characteristic of this transducer to be linear and pass through zero, the initial gap between the light-absorbing layer and the upper face (base) of the prism must be approximately 0.3 μm.

Figure 2 shows computed dependence of the OFPT output signal in relative units as a function of membrane movement w. Since membrane movement is linearly dependent upon the change in pressure, the OFPT's steady-state characteristic (output signal as a function of pressure) will take a similar form.



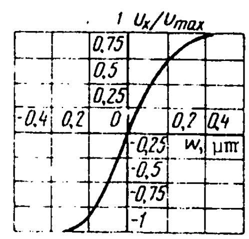


Figure 1. Optical-fiber pressure transducer with destruction of TIR: 1 - membrane; 2 - light-absorbing layer; 3 - - seal; 4 - fiber light guides; 5 - photosensitive bridge circuit; 6 - LED; 7 - - housing; 8 - prism; 9 - cover.

Figure 2. Steady-state characteristic of OFPT with solid light-absorbing layer: U_x, U_{max} - actual and maximum output signal values respectively.

The range of membrane movement lies within the limits of 1 μm. These small movements give this pressure transducer its high speed on the one hand, but on the other such a narrow gap between the two surfaces imposes very high requirements for precision in fabrication and assembly. The membrane and light-absorbing layer in this instance should be made of glass, a material making possible the necessary degree of precision.

A gap of the order of a fraction of a micrometer can be obtained, for example, by applying a thin ring-shaped film to one of the surfaces using the process employed in applying antireflection coatings to optical parts.

Precision requirements can be substantially reduced if a plastic material, silicone, for example, is used for the light-absorbing layer rather than the glass. Despite the fact that this narrows the temperature range, the latter nevertheless remains broad enough since the elastic and optical properties of silicone vary only slightly within the temperature range of -60 to +350°C [6].

The steady-state characteristics in Figure 3 are those of transducers with light-absorbing layers made of silicone with a refractive index of 1.57 and an absorption index of $5 \cdot 10^2$ cm^{-1}. Since the depth of irregularities of the rubber used were approximately 20 μm, the range of membrane movement was also roughly 20 μm. The broken line in Figure 3 shows the steady-state transducer characteristic for this case. It

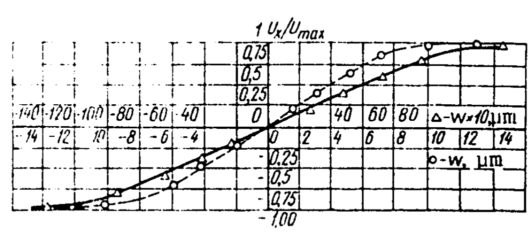


Figure 3. Experimental steady-state characteristics of OFPT with plastic light-absorbing layer.

Movement of the membrane in this instance varied not the width of the gap, but the area of (optical) contact. Irregularities in the silicone took the form of a grooved surface with saw-toothed cross section. In the case of this type of irregularity the theoretical dependence of the output signal in relative units upon membrane deflection w is described by the expression $U_x/U_{max} = \frac{2}{h} w$, where h - depth of irregularity.

The fabrication and assembly of OFPT with destruction of TIR would be still further simplified if the light-absorbing silicone coating were shaped, for example, as a semicylinder or a prism with its apex pointed toward the upper face of the prism and attached by its plane side to the membrane.

Movement of the membrane will deform the semicylinder (prism), which will change the area of contiguity between the plastic material and the upper face of the prism (area of optical contact). In this case the value of the output signal in relative units will be computed in accordance with the formula

$$U_x/U_{max} = \frac{2\int_0^{\varepsilon}\left[\int_0^{\varphi(x)} I(x, y)\,dy\right]dx}{\int_0^{d}\left[\int_0^{\varphi(x)} I(x, y)\,dy\right]dx},$$

where $I(x, y)$ - distribution of intensity of radiation from fiber in plane of optical contact; $\phi(x)$ - analytical expression of cross section of fiber light guide as plane of optical contact; $\varepsilon = f(w)$ - area of optical contact in the form of a function of the deformation of the plastic layer; d - cross sectional area of fiber light guide as plane of optical contact.

Linearity of steady-state characteristic with the given value for fiber diameter is achieved by changing the form or radius of curvature of the light-absorbing plastic coating.

The solid line in Figure 3 represents the characteristic of an OFPT with a silicone coating in the form of a cylinder halved along its axis and having a radius of curvature of 2 mm and a segment height of 1 mm; the cross sections of the fibers, arranged at an angle of 45° with respect to the base of the prism, are in the form of a circle 2 mm in diameter.

The OFPT with destruction of TIR described here is distinguished to advantage from transducers with reflecting membranes and moving reflectors by the fact that the linearity of their characteristic curve does not depend upon the dimensions of the housing or light guides or upon the distance between the membrane and the light guides, which, first, permits substantial simplification of the computation and design of OFPT and, second, makes possible a reduction of optical losses, since the light guides may be brought as close as desired to the modulated surface of the prism without degradation of the linearity of the curve. Moreover, with respect to speed OFPT with destruction of TIR exceed virtually all other pressure transducers because of their exceptionally narrow range of membrane movement.

BIBLIOGRAPHY

1. Pahler, R. H. and Roberts, A. S. Trans. ASME, 1977, Vol 99, No 1.

2. Application 2450080 (FRG).

3. Pat. 3789667 (USA).

4. Poddubnyy, S. A. "Quartz Glass as a Material for Elastic Components and a New Technology for Precision Quartz Work." In "Perspektivy razvitiya uprugikh i chuvstvitel'nikh elementov" [Prospects for the Development of Elastic and Sensitive Components], Moscow: TsINTI elektrotekhnicheskoy promyshlennosti i priborostroyeniya, 1961.

5. Udalov, N. P., Busurin, V. I. and Pasynkov, V. I. "Izmeritel'nyye ustroystva na osnove optrona s upravlyayemym kanalom odnogo tipa" [Optron Measuring Devices with a Single Controled Channel]. Moscow: MDNTP, 1978.

6. Andrianov, K. A. "Teplostoykiye kremniyorganicheskiye dielektriki" [Heat-Resistant Silicone Dielectrics]. Moscow: Energiya, 1964.

8963
CSO: 1863/202

GRAPH PLOTTERS WITH LINEAR ELECTRIC DRIVE

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 5, May 81 pp 23-24

[Article by D. V. Svecharnik, doctor of technical sciences and Yu. M. Osipov, engineer: "Graph Plotters with Linear Electric Drive"]

[Text] Linear electric drive has recently begun to be employed in instruments and industrial equipment [1-3]. The operation of these devices on inclined planes of motion under the impact of external mechanical perturbations (vibrations, rolling, shocks) presents developers with a number of new problems, the basic one being that of determining stability criteria for linearly moving drive components making possible the achievement of a desired precision.

The authors have proposed a graph plotter with linear electric drive [4] in which the linearly moving components, the inductor 1 and recorder 2, are "held" away from external mechanical effects by the inertia of their counterbalanced masses (see figure). Stability conditions have been computed for linearly moving drive components in the form $\Sigma F_{iY} = 0$, $e_{iXZ} = 0$ where F_{iY} - vectors of the forces acting on drive components along axis Y; e_{iXZ} - eccentricity between the centers of gravity of the components and axes X and Z.

A certain degree of error is involved in the satisfaction of these conditions in practice since all machine counterbalancing methods currently employed are intended to approximate these conditions more or less. Permissible imbalance $\Delta m = m_{ЛД} - m_{УР}$ may be calculated from the equation for movement of the linear drive for coordinate l of the graph plotter:

$$(m_{ЛД} + m_{УР}) \frac{d^2 l}{d t^2} + \left| m_{ЛД} - m_{УР} \right| g_{Г} \sin\alpha + (m_{ЛД} + m_{УР}) \times$$
$$\times g_{Г} \left[f_{ТС} \left(\frac{d}{D} + \frac{4 d_{Р}}{D_{Р}} \right) + \frac{4 \xi}{D_{Р}} + \frac{2k}{D} \right] \operatorname{sign} \frac{dl}{dt} +$$
$$+ 2 l_0 \varepsilon \left(\frac{1}{D} + \frac{2}{D_{Р}} \right) + k_{ВТ} \frac{dl}{dt} + D_{\psi} \frac{dl}{dt} + F_{ДВ} = 0,$$

where $m_{ЛД}$ - mass of linear motor inductor; $m_{УР}$ - counterbalancing mass (recorder etc.); $g_{Г}$ - acceleration of gravitational forces; α - angle of incline of linear movement with respect to horizontal plane; $f_{ТС}$ - coefficient of sliding friction; d - diameter of inner ball bearing ring; D - external ball-bearing diameter; $d_{Р}$ - diameter of belt-drive roller journal; D_p - roller diameter; ξ - coefficient of belt rigidity; k - coefficient of rolling friction; I_0 - moment of inertia of roller rotation; ε - angular acceleration of ball bearings and rollers; k_{BT} - coefficient of internal friction; D_ψ - coefficient of electromagnetic damping of motor; $F_{ДВ}$ - driving force of linear motor.

We will assume for the steady-state counterbalancing process a vertical line of drive-component movement, an acceleration of external perturbing forces w_a equal to 0 and cut the power off to the linear motor. In this case, damping force $D_\psi \frac{dl}{dt} = 0$, while internal friction may be disregarded. Inertial forces $2 I_0 \varepsilon \left(\frac{1}{D} + \frac{2}{D_p}\right)$ are small as compared with available forces. After conversion we obtain $\Delta m \leqslant 2m_{yp}A:(\sin\alpha - A)$, where $A = f_{TC}\left(\frac{d}{D} + \frac{4d_p}{D_p}\right) + \frac{4\xi}{D_p} + \frac{2k}{D}$ - coefficient of component resistance numerically equal to the angle of friction.

From this expression it can be seen that component movement is possible when $\sin\alpha > A$. Permissible imbalance with vibration and rolling

$$\Delta m \leqslant \frac{2m_{yp}A\,(g_\Gamma + w_a \sin\beta)}{(g_\Gamma + w_a\cos\beta)\sin\alpha - A\,(g_\Gamma + w_a\sin\beta)};$$

here β - angle of action of acceleration $w_a \approx \frac{S_a}{250} f_B^2$ [5], where f_B - frequency of vibration and rolling; S_a - amplitude of oscillation.

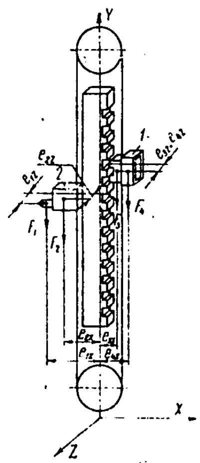
An experimental model was constructed with a linear electric drive for X coordinate, the linear motor in which is a three-phase electric machine with an interdigital magnetic circuit. Tests for prolonged inclination and vibration stability on an ST5000/300 vibrator in various modes of linear motor operation have demonstrated the efficiency of this method of counterbalancing linearly moving components of a coordinate of an automatic recorder with accelerations of vibrational oscillations to 3.2 g_Γ.

An expression has been obtained for computing magnetic conductivity in the air space between the interdigital inductor and the secondary component with the two in varying relative positions:

$$G = \frac{1}{\delta}\left[1 - \frac{B\,(f - B)}{g + 9B\left(f - \frac{1}{3}S_{13}\right) - C}\right]$$

where δ - size of air space: $B = 16h^2/g\pi^2\delta$; $C = 20h^2B/\pi^2\delta$ - coefficients for simplification of recording (h - width of space on interdigital surface); f, g and S_{13} - coefficients presented in [3].

A FORTRAN-language program was written in accordance with this algorithm; a YeS-1022 was used to find computed values and curves were plotted of the values of conductivity as a function of the shift of interdigital inductor and secondary-component surfaces $G = f(\xi)$. On the basis of this family of curves were selected for obtaining maximum modulation $(G_{max} - G_{min})/G_{max}$ and the maximum value for $dG/d\xi$, which determines maximum electromagnetic force.



Kinematic diagram of linear drive of graph-plotter coordinate: F_1 - F_4 - vectors of forces acting on drive component along axis Y; e_{1Z} - e_{4Z}, e_{1X} - e_{4X} - eccentricity between component centers of gravity and Z and X axes respectively.

Use of a linear electric drive as proposed in the description in [4] has made it possible to simplify graph-plotter design, reduce the amount of auxiliary equipment required (compressor, pneumatic system etc. [1, 2]) and metal consumption (due to absence of solid ferromagnetic platform [1, 2] and improve reliability indicators: probability of failure-free operation has been increased 1.2-1.5-fold, average period of failure-free operation 1.4-1.7-fold. The linear electric drive may be mounted on a movable for operation in an environment of external mechanical perturbations, which substantially expands its sphere of application in contemporary instruments and industrial equipment, in the manufacture of robots for example.

The manuscript of this article has 11 pages of text, 5 figures, 1 table and a bibliography of 5 entries; it is cataloged as No. 1487 in TsNIITEI priborostroyeniya's [Central Scientific Research Institute of Information and Technical and Economic Research on Instrument Manufacturing] collection of deposited works.

BIBLIOGRAPHY

1. "Components, Assemblies and Equipment," "Byulleten' inostrannoy nauchno-tekhnicheskoy informatsii TASS" [TASS Bulletin of Foreign Scientific and Technical Information], 1973, No 1, p 1568.

2. Belyavskiy, Ye. I., Lyashchuk, Yu. V., Onegin, Ye. Ye. and Yarosh, A. V., "The Manufacture of Linear Stepping Motors," ELEKTRONNAYA PROMYSHLENNOST', 1979, No 3.

3. Svecharnik, D. V., "Distantsionnyye peredachi" [Long-Distance Transmissions], Moscow: Energiya, 1974.

4. Inventor's Certificate 553451 (USSR).

5. Iorish, Yu. I., "Vibrometriya" [Vibration Measurement], Moscow: Mashgiz, 1963.

8963
CSO: 1863/202

UDC 681.327.63.004.1

EXTERNAL MEMORY WITH RANDOM ACCESS TO AN AUTOMATED CONTROL SYSTEM

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 3-5

[Article by engineers A. V. Spirkov, V. N. Telezhenko, A. A. Podgornyy and A. A. Alfimtsev]

[Text] External memories with random access, which assure the storage of large volumes of information and rapid access to it in solving tasks of the user are an inseparable part of control problem-oriented complexes.

Modern external memories are improved by increasing their capacities, speed and reliability, on the one hand, and also by reducing their dimensions, mass and cost per bit of stored information, on the other, which on the whole is achieved by applying new circuit solutions and a new elementary base, a more improved technology for the manufacture of the external memories themselves and information carriers, and also by the creation of means of self-diagnosis.

Together with very dynamically developing flexible magnetic disk stores and rigid irremovable disk stores, standard plug-in packet disk stores are finding wide application.

The proposed SM-5407 plug-in magnetic disk external memory, intended for the storage of large volumes of information and its operative input and output, was made with the requirements of high reliability, compactness, standardization, operating convenience, maintainability and self-diagnosis taken into consideration. In the creation of the plug-in magnetic disk external memory experience was used which was acquired in the development of the A3283 plug-in magnetic disk stores control device, which has recommended itself with users as a highly reliable part and one convenient to operate. In particular, a series approach has been preserved toward diagnosis: after the internal test programs of the device have been run, its reliable functioning under the control of the system programs is guaranteed. It is important that use of a computer complex is not required for diagnosis of the plug-in magnetic disk external memory.

The macroarchitecture of the plug-in magnetic disk external memory is determined mainly by 12- and 16-digit program-addressable registers, the instruction code and the interfaces for communication with the stores of the complexes. Each program-addressable register carries its own functional load in part of the monitoring of the operational system of the state of the plug-in magnetic disk external memory equipment, the sequence of data transmission and the performance of instructions issued by the central processor. All the program-addressable registers are accessible both for the system and for the panel mode unit of the device.

Technical Characteristics of Plug-in Magnetic Disk External Memories

Maximum number of connected YeS-5061 stores	Up to 8
Type of information carrier	YeS-5261 disk packet or packet compatible with it
Data transmission rate in words per second	156K
Disk packet capacity in 1000 words	10
Data storage format	Sectoral
Number of:	
cylinders	203 (including 3 spares)
heads	20
sectors	10
words per sector	256
Data words capacity	16

Instructions executed by plug-in magnetic disk external memories are divided according to functional purpose into preparative and executive. The execution of preparative instructions does not involve data transmission and therefore the simultaneous working of several stores can be assigned. Execution of executive instructions involves data transmission. Those instructions transfer the plug-in magnetic disk external memories into an occupied state for all the time--from the start of execution of the instruction to its completion. In that case the functioning of the device cannot be halted for the execution of any other instructions.

The microarchitecture of a plug-in magnetic disk external memory is determined by a 40-digit word of a microinstruction and the composition of instructions of the series K589 microprocessor complex. The microinstruction word consisting of 12 controlling fields assures control of the functional units of the operational part of the device during the execution of instructions issued by the system's central processor. The composition of the instructions of the microprocessor complex consists of flexible means of processing the internal information of the operational part of the device for formation of the addressing sequence during the running of microprograms.

It takes very little additional hardware to provide the possibility of referring to the internal registers of the microprocessor, made of elements of series K589, on the part of both the system's central processor and the panel mode units of the device.

The SM-5407 device has the following characteristics:

--the possibilities: a) of connection to the two independent computer complexes SM-3 and SM-4; b) of work under conditions of normal and rapid transmission (during extraprocessor data transmission); c) testing without an external store of the efficiency of the controller equipment by running test programs with the use of programmable registers simulating a small interface (an external memory interface); d) of work with disk packets without sector recesses; e) the possibility autonomous initialization (marking) of a disk packet;

--standardizations in the design decisions and also in the scheme for realization of the operational part and output on a "common bus" interface;

--architectural continuity of the device during connection of other external memories (for example, magnetic disk stores with a capacity of 100 Mbytes);

--microprogram realization, which assures convenience of adjustment and operation, and also the possibility of modifying the execution of instructions.

The programming of input/output operations in the system is performed by nine program-addressable registers. The remaining three program-addressable registers are monitoring registers and permit simulating the work of external memories on the part of the external memory interface. The connection of the plug-in magnetic disk external memory to two independent complexes makes it possible to economize in the application of additional external memories in a regime of distributed data processing.

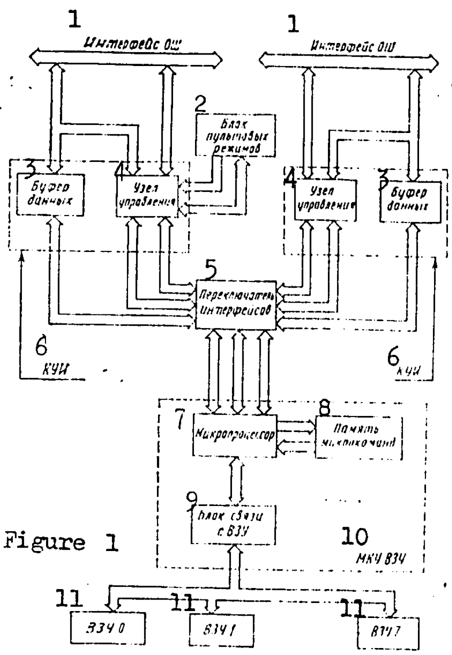


Figure 1

1 - Common bus interface
2 - Panel mode unit
3 - Data buffer
4 - Control unit
5 - Interface switch
6 - Interface controller
7 - Microprocessor
8 - Microinstructions memory
9 - External memory communications unit
10 - Microprogrammed controller of external memory control
11 - External memory 0, 1, 7

Figure 1 presents a block diagram of the device. The plug-in magnetic disk external memory includes a controller of interface control, a microprogrammed controller of external memory control and the external memory. The block structure used in the development of the controllers permits standardizing separate logic units in constructing devices for a similar purpose.

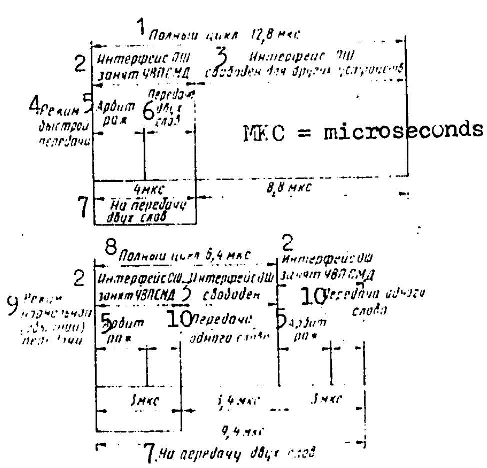
The controller of interface control is intended for connection of the device to the interface of a computer complex and assures the possibility of data transmission upon interruption of the processor and direct access to the memory, and also connection of the device to the panel mode unit. By means of a standard set of signals which is readily realized in a portion of the microprogrammed controllers, connection of the controller of interface control with the microprogrammed controller of external memory control is assured. About 80 percent of all the external memory units consist of constructively and electrically developed units realized in such a manner that they can be used in creating other external memory devices.

The standardization of such a level is achieved in part because in the controller of interface control registers are derived which are common to most known external memories. In that case, as is well-known, the problem arises of "translating" the addresses of registers. In the proposed development, address translation is solved by means of bridges. The possibility of "translating" the addresses of program-addressable registers by means of a semiconductor memory also is being studied.

An unquestionable achievement of the interface control controller is the presence of a data buffer with a variable depth of bufferization (the depth of bufferization can be 16, 32 or 64 data words), satisfying in speed a broad class of devices. The data buffer is mainly necessry for extra-processor data transmission, the control circuit of which assures functioning of the device in two transmission modes: the normal (ordinary) and rapid (as desired by the user). During work in the rapid mode, two data words are transmitted per cycle of common bus interface capture. If the number of transmitted words is odd, then during functioning of the device in the rapid mode the last data word is transmitted in the normal mode. Such a mode of operation permits curtailing the time of common bus occupancy and makes it possible for other devices to obtain access to the bus during its liberation.

In the described device the delay for double bufferization with the rate of rotation of the disk packet taken into account is 12.8 microseconds. Thus the controller of interface control must return to the common bus every 12.8 microseconds and the transmission time correspondingly must not exceed 12.8 microseconds. As Fig 2 shows, 4 microseconds are expended by the common bus on the transmission of the first two words; 2 microseconds each on arbitrage and the transmission properly speaking of two data words (1 microsecond per word), that is, in the course of 8.8 microseconds (12.8 - 4) the other devices can have access to the common bus. During the transmission of a single data word (the last word when there is an odd number of words) the total time cycle amounts to 6.4 microseconds. Thus under normal data transmission conditions only 3.4 microseconds (6.4 - 3) will be presented to other devices (2 microseconds go for arbitrage in each transmission cycle).

Figure 2



1 - Total cycle, 12.8 microseconds
2 - Common bus interface is occupied by plug-in magnetic disk external memory
3 - Common bus interface is free for other devices
4 - Rapid transmission mode
5 - Arbitrage
6 - Transmission of two words
7 - For transmission of two words
8 - Total cycle, 614 microseconds
9 - Normal (ordinary) transmission mode
10 - Transmission of one word

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

The noted features of the controller of the interface control permit creating high-capacity computer complexes with a large number of peripherals.

In the development of the controller of external memory control the goal was pursued of constructing an economical, simple and also powerful in part of the equipment and operating speed microprogrammed device which permits, with great reliability, determining the possibility of functioning of the device as a whole without loading the computer complex. At the present time the microprocessor set of series K589 is the most accessible and satisfactory with respect to all the enumerated requirements.

On the basis of available experience a 16-digit standardized microprogrammed controller with a 40-digit microinstruction word has been created which has permitted programming the microprocessor with a relatively small number of simple microinstructions. Additional equipment not included in the microprocessor set and necessary mainly for organization of the connection of the microprogrammed controller of the external memory control with the controller of the common bus interface control and with the external memory, was required in a minimum volume in that case.

The unit for connection with the external memory is constructed with consideration of the specifics of the individual external memory. By means of that unit the microprogrammed controller of external memory control is readily adapted to a specific external memory with not more than 36 interface lines.

The structure and architecture of the external memory must satisfy the requirements of the minimum expenditure of time on the search for defects. The effectiveness of use of an external memory in an automated control system is a definite function of the means of diagnosis built into the design in the process of planning the article.

The microprogrammed realization of the controller of external memory control permits observing the following basic principles of construction of means of diagnosis of peripherals:

--a minimum of additional built-in equipment (of the total volume of equipment) for the diagnostic procedures;

--a maximum of inclusion of all component parts of the device by diagnostic procedures;

--the possibility of assigning critical modes of operation to test the means of protection of the device;

--the possibility of using automated methods of planning means of diagnosis;

--the connection of means of diagnosis with means of monitoring information transmission;

--location of a defect precisely to the replaceable element.

The technical state of a plug-in magnetic disk external memory can be analyzed by combined means of diagnosis: programmed, built-in, panel and external. Programmed means permit giving functional tests in order to test the general efficiency of functional units and subassemblies of an article, and also of circuits for monitoring information transmission during the execution of instructions to record and read. Built-in devices are designed for giving diagnostic tests of the operational part of the equipment and external memory. Defects of the operational part are located with precision to the element. Panel equipment is intended for assigning effects to the nucleus of the operational part of the device, with indication of the result of the processing of those effects by the equipment. Testing of the equipment registers is provided by the recording and reading of various kinds of code combinations. In addition, effective debugging of the device without external memories is assured by programming the control program-addressable registers described in the article. The external devices offer the possibility of assigning debugging modes of operation of the device with visual observation of the time correlations of signals on an oscillograph screen.

2174
CSO: 1863/210

UDC 681.327.28

RAISING RELIABILITY OF SEMICONDUCTOR READ-ONLY MEMORY

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 5-6

[Article by V. Ya. Yurchishin, engineer, in the section "Automated Control Systems," "Development and Use of Peripherals in ASU's," (Thematic Selection, continued from No 5, 1981)]

[Excerpts] It is important to raise reliability without resorting to a substantial increase in apparatus outlays and a reduction in response speed. Meanwhile, the wordlength used in the small computer system and especially in the peripherals is 8 to 16 bits, which makes it difficult to satisfy this requirement.

Since the failure rate of connected elements (in the load state) so substantially exceeds that of unconnected elements that the failure rate of the latter may be ignored, and since PZU [read-only memory = ROM] is a nonvolatile device, it is expedient to consider a ROM structure from the position of achieving a possible shift of all or part of its equipment from the load to the no-load state, i.e. of reducing the time the ROM has a load on it. This is possible, for example, if the voltage supply is connected to the device or its individual parts for only that part of the machine cycle sufficient to extract information from the ROM.

All three methods of ROM division are finding practical application. As a consequence of this, not only is reliability enhanced, but power consumption is substantially reduced too.

It is most expedient to use the described principle for raising reliability in ROM based on LSI TTL with fusible links which have the properties of random restoration of the burned out links and increased power consumption.

Shown in fig. 3 [not reproduced] is a structural diagram of this type of device. Suppose a 16K-word ROM has a $\lambda = 10^{-3}\ h^{-1}$. Then the reliability for 1,000 hours of operation under continuous load will be $R(1000) = \exp(-\lambda T) = 0.368$.

For a device of the same size, made in accordance with fig. 3 with $\tau/T_{cycle} = 1$ and capacity of the microcircuits $N_{DSC} = 256 \times 4$, the same reliability is achieved for 64,000 hours. With that, power consumption is lower by a factor of 10. The validity of the derived reliability characteristics has been repeatedly confirmed by experiments.

For LSI ROM's with fusible links, it has been noted that the failure rate (time to restoration of the burned out links) is a function of the supply voltage. If its value is greater than rated, the failure rate increases. A sharp increase in the failure rate has been observed when the supply voltage is raised to +6V.

This peculiarity led to accelerated comparative tests of LSI ROM's with fusible links with a variable and constant load time. Two devices of identical size, 16K bytes, in which identical information had been recorded were selected for the tests. One device was connected to a circuit with a constant load, the other --in accordance with fig. 3. The tests were conducted over a period of 300 hours. Supply voltage levels in both devices were identical (the voltage drop in the voltage pulse shapers was compensated for). The devices used were LSI ROM's with fusible links, programmed on a series programmer and subjected to four hours of pre-burn-in.

Within 50 hours after the start of the tests, failures--restorations of fused links--began to appear in the constantly loaded LSI ROM. Eight restorations had occurred in this ROM by the end of the test (300 hours). The device with the lowered load withstood the test without a single failure.

Similar tests on two 32K-byte devices showed 13 failures in the constantly loaded ROM and none in the device made in accordance with fig. 3. Tests on smaller ROM's also confirmed that reducing the load on the ROM leads to an increase in its reliability.

Semiconductor ROM's made in accordance with the diagram shown in fig. 3 are used as microprogram memory in VZU [external storage units]: the SM-5407 with removable magnetic disks and the SM-5003 magnetic tape unit.

8545
CSO: 1863/217

UDC 681.327.64

MASS EXTERNAL STORAGE WITH SEQUENTIAL ACCESS IN AN AUTOMATED CONTROL SYSTEM

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 1-3

[Article by engineers V. V. Kovbas, V. K. Dragunov, V. M. Zuyko and V. I. Kovbasenko, in the section "Automated Control Systems," "Development and Use of Peripherals in ASU's," (Thematic Selection, continued from No 5, 1981)]

[Excerpt] Complexing of magnetic tape external storage units (UVPML) is done by including a nomenclatural UVPML, varying in the number of NML [magnetic tape storage units] used, in the UVK [control computer complex] (for example, using the A3116 UVPML in the M-6000 UVK and the SM-5301 UVPML in the SM-4 UVK). Also included in the control computer complex are a nomenclatural NML controller and a varying number of nomenclatural NML's equipped with a cable for communication with the NML controller (for example, use of the A3181 NML controller and the YeS-5012 NML in the M-4030 and M-4030-1 control computer complexes). Developmental and operational experience shows that UVPML complexing must follow the modular principle by including in the control computer complex a nomenclatural NML controller and the needed number of nomenclatural NML's.

Development of incorporated NML's and an NML controller will yield certain advantages during operation of them in many systems.

The use in the first phase of SM computers of NML's having a different interface with the controller has led to difficulties in development of a controller allowing connection of the NML's of the entire spectrum of the first phase of the SM computer system to the SM-3 and SM-4 control computer complexes. Second-phase magnetic tape storage units will be produced only with a unified interface. Series production of the SM-5301 UVPML has now begun for the SM-3 and SM-4 control computer complexes; use of this UVPML allows connection of one to four IZOT 5003 or SM-5300 type NML's and also supports interexchange of recorded mangnetic tapes with a 32-bit/mm recording density among the SM, ASVT-M and YeS computer systems.*

* Kovbas, V. V.; Kulik, A. I.; and Ozhiganov, Yu. M., "Organization of External Storage with Sequential Access in the Small Computer System (SM EVM)," PRIBORY I SISTEMY UPRAVLENIYA, No 5, 1979.

In the aggregate, each UVPML developer solves three problems: reduce cost, increase reliability and expand the functional capabilities of UVPML's. Any engineering design is a compromise solution to these problems. The operating experience of the A3181, A3116 and SM-5301 units in various ASU's indicates that these are highly reliable units that are convenient to operate. However, their hard wiring has prevented expanding the diagnostics in the NML controller; in the final analysis, the availability of this capability would allow raising the operational reliability of the units. Reliability and repairability are two very important characteristics of many ASU's. They determine the capabilities of the system from the position of solving its basic problem--control.

To improve the technical and operational characteristics of control computer complexes that have the "Common Bus" (OSh) interface, the second phase of the small computer system will use the SM-5003 magnetic tape unit controller; this controller will allow connecting to a control computer complex the SM-5305 magnetic tape unit or any other SM computer magnetic tape unit that has a 2 or 1 m/s magnetic tape speed.

Use of the FK [phase encoding = PE] recording method allows increasing recording density on magnetic tape, i.e. increasing the capacity of external storage and the rate of operation of the control computer complex. With the PE recording method, one reel of magnetic tape holds 40 Mbytes.

Basic Characteristics of the SM-5003 Magnetic Tape Unit Controller

Maximum number of SM-5305 tape storage units that can be connected:		up to 4
Maximum data transmission rate in words per second:		63K
Recording method:		PE and NRZ
Recording density in bits/mm:	with PE	63
	with NRZ	32

With the BVN-1 [non-return-to-zero = NRZ] recording method, ones and zeros are represented by a change or absence thereof, respectively, in the condition of tape magnetization, while with the PE method--by a change in condition of magnetization. In the process, if the bit value on a given track matches the value of the preceding bit, for example, 11 or 00, the magnetization sense changes twice; if the values of the preceding and following bits differ, the magnetization sense changes once.

8545
CSO: 1863/217

UDC 621.396.66.004.13

SYSTEM FOR AUTOMATIC MONITORING OF A DATA TRANSMISSION DEVICE

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 8-9

[Article by engineers V. N. Kiselev, V. V. Apanasenko, and V. S. Blagoveshchenskiy]

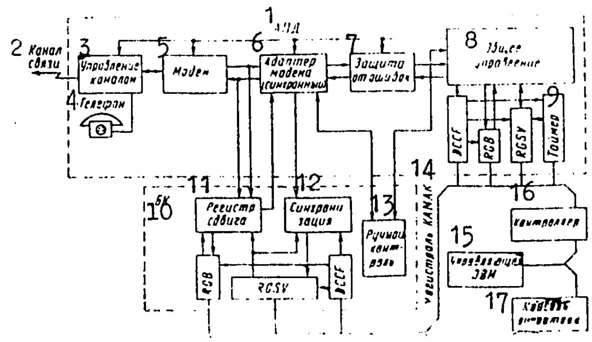
[Text] The most important characteristic of an APD [apparatura peredachi dannykh--data transmission device] is reliability of data transmission. It determines the effectiveness of operation of computer systems. Therefore monitoring the work of the APD is necessary both in the stage of its manufacture and debugging and in the process of operation. To enhance the quality of information transmission, in a number of cases devices are used for protection against error which do not always guarantee high quality of work of the APD, and in a number of cases, for example, in the transmission of information about physical experiments, it is advisable to create a system for autonomous monitoring of the APD.

The reliability of functioning of an APD is determined by the effectiveness of the used methods of operative monitoring. The monitoring procedure must be organized so as not to allow possible malfunctions and other disturbances in the work of the equipment.

The structural diagram of an APD, made in accordance with requirements of the CAMAC standard [1,2] and the described system for its monitoring, is presented on the figure. The system of autonomous APD monitoring includes a control computer (an M-400 in the case under consideration), an operator's console, a controller and a monitor unit and assures the execution of program and apparatus monitoring of all the main APD assemblies and units.

During programmed monitoring a loop is organized (an information loop) between the APD being tested and the computer; it permits dynamically testing the efficiency of APD assemblies and units in "Reception" and "Data Transmission" modes. Organization of the loop provides a monitor unit, which is connected with the APD, on the one hand, and the control computer, on the other. During testing of efficiency in the "Reception" mode the monitor unit synchronously receives test combinations from the computer and transmits them in sequential code to the APD input. In testing the apparatus in the "Data Tranmission" mode the monitor unit synchronously receives information from the APD output and transmits received code combinations in parallel code to the computer.

The monitor unit is connected to the APD by the synchronization assembly and shift register. The former generates cyclic pulses which assure synchronous reception and transmission. Exchange of data in a sequential code occurs through the shift register.



1 - APD
2 - Communications channel
3 - Channel control
4 - Telephone
5 - Modem
6 - Modem adapter (synchronous)
7 - Error protection
8 - General control
9 - Timer
10 - Monitor unit
11 - Shift register
12 - Synchronization
13 - Manual control
14 - CAMAC main line
15 - Control computer
16 - Controller
17 - Operator's console

The monitor unit is connected with the CAMAC main line through an interface, the composition of which includes an instruction decoder CDDF, a service register RGSV and a data buffer register RGB.

The work of separate assemblies of the apparatus (registers, decoders, a programmed timer, etc) can be tested by the programmed monitor.

The software of a system of autonomous APD monitoring is constructed on the modular principle and contains a number of test programs, each of which monitors one parameter of the apparatus. The composition of the programs includes auxiliary subroutines included directly in the test programs using them, assuring the preparation and adjustment of the entire system (see figure) for conducting tests. The system software includes a program monitor intended for the selection and issuance of any program from the list of tests. The language of directives is used as the language of communication of the user with the system. The controlling directives introduced into the computer through the operator's console with a "Consul" electric typewriter are received and analyzed by the program monitor. By means of a directive a test number and the monitoring mode (single or multiple) can be assigned. Upon conclusion of reception of the directive the monitor adjusts and starts the selected test program.

In the process of the work of test programs when necessary on the "Consul-260" typewriter instructions are derived which require definite actions from the operator for the conducting of further testing. Upon conclusion of the work the test programs anslyze the information obtained in the course of the test and issue the monitoring results to the operator.

Apparatus monitoring of the APD is accomplished by means of a manual control unit which is a part of the monitor unit. The manual control permits testing the efficiency of the apparatus by cycles and monitoring it in greater detail than with the programmed monitoring.

The monitor unit, which is a part of the autonomous monitoring system, consists of a module made in standard designs of the CAMAC system.

BIBLIOGRAPHY

1. Kiselev, V. N., Apanasenko, V. V., and Davydov, Yu. N. "Apparatus for Transmission of Data of an Experimental Data Collection System." In the collection: Tekhnicheskoye obespecheniye i zadachi vychislitel'nogo tsentra kollektivnogo pol'zovaniya (Hardware and Tasks of a Collective-use Center). Tomsk, Izd-vo TGU, 1977.

2. Apanasenko, V. V., Kiselev, V. N., and Davydov, Yu. N. "Principles of Construction of Data Transmission Apparatus Software." Ibid.

2174
CSO: 1863/210

UDC 62-229.72

INDUSTRIAL MINI-ROBOTS OF MODULAR CONSTRUCTION WITH DISCRETE PNEUMATIC DRIVE

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 22-23

[Article by engineers D. L. Krolik, S. T. Khanukayev, E. A. Kurman, E. A. Onishchenko, A. D. Abitov, and M. N. Poyda]

[Text] At the Design and Technological Bureau (Nal'chik) a series of industrial mini-robots with pneumatic two- or multi-range drives, operating with custom-made parts with a mass of up to 0.5 kg, have been developed. A corresponding combination of modules as a function of requirements of the technological process permits obtaining robots which work with intermediate positioning or with positioning at two extreme points. In that case the precision of positioning in both cases is high enough (± 0.05 mm).

It is far simpler to obtain the control system of a robot with intermediate positioning than systems serving similar purposes and having as the object of control hydraulic and electric servodrives, including with step motors.

The mini-robots under consideration can be made up of five main modules: two-range, multi-range, elevation, rotation and transverse displacement (support). Each module is an independent functional unit which has its own pneumatic drive, and if necessary also pneumatic distributors, position sensors, dampers, valves, etc. The pneumatic drives work on compressed air with a pressure of at least 0.35 MPa (3.5 kg-force). Shown on Figure 1 is a diagram of the PMR-0.5-254PV pneumatic mini-robot, composed of the first four modules.

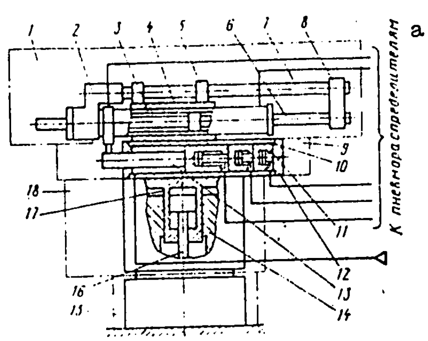


Figure 1. Diagram of the PMR-0.5-254PV mini-robot.
a: to pneumatic distributors

Two-range module 1 is made in the form of a pneumatic cylinder 3 with the possibility of movement in casing 4, where brakes are installed to fix the pneumatic cylinder in the necessary position along the axis. The stroke of rod 6 of the pneumatic cylinder is 128 mm. The cylinder is also filled with a rod stroke of 200 mm. The rod is connected by the anterior 2 and posterior 8 cross-piece with stem 7, which moves in guides 5. The stem prevents rotation of the rod around the axis. On the anterior end of rod 6 there is a seat for fastening the clamp or wrist of the industrial mini-robot. On each cross-piece a hydraulic damper is fastened for deceleration at the end of the stroke back and forth. The action of the damper is regulated by a set screw.

In the case of independent application the two-range module permits composing industrial mini-robots for positioning the clamp at two points with a stroke of the latter of 128 or 200 mm.

The multi-range module 9 represents a pneumatic cylinder 10 in which six complex pistons are set, five of them with shanks (Fig 1 shown three pistons). The first complex piston is fastened by projections behind the shank of the cover 12, the next one behind the shank of the preceding one, etc. The sixth piston is connected with the rod 13 which emerges through the cover. The pistons from the first to the sixth have strokes of 2 (2^1), 4 (2^2), 8 (2^3), 16 (2^4), 32 (2^5) and 64 (2^6) mm respectively. Thanks to the constant feeding of compressed air into the rod cavity all the pistons are shifted into the extreme right position. During the feeding of compressed air from the pneumatic distributor under any of the pistons the sixth piston is shifted to the left by the amount of the stroke of that piston toward which the air was fed, on account of the difference of pressures in the piston and rod cavities. By thus feeding compressed air to all the pistons simultaneously or in a definite combination it is possible to obtain a pneumatic cylinder rod stroke of 126 mm with a discreteness of 2 mm. The rod of the sixth piston has an opening for a pin fastened on the pneumatic sylinder of the two-range module. The combination of two- and multi-range modules gives a summary stroke of the clasp of 254 mm with a discreteness of 2 mm.

By adjusting the washers placed between the pistons the stroke of each piston is assured with an error of about 0.01 mm during a stroke up to and over 8 mm respectively. The summary error, tests have shown, is in the range of $\pm$ 0.05 mm.

The elevation module 18 is made in the form of a vertical pneumatic cylinder 17, which has at the top a developed flange for fastening the arm of the industrial mini-robot. The pneumatic cylinder can move in the guides of a post 14 which is the casing of the module. Piston 16 is fastened in the post. When air is fed into the pneumatic cylinder (not shown on Fig 1) the casing of the pneumatic cylinder 17 moves, with the arm fastened to it. Brackets with pneumatic distributors are fastened to the pneumatic cylinder flange.

The rotation module 15 represents a geared shaft with a flange, rotating in bearings set in the casing. The elevation module and arm of the industrial mini-robot are mounted on the flange. Placed in the same casing is a pneumatic cylinder with a strip which interacts with the geared shaft and rotates the flange around the vertical axis. The rotation module assures the work of the industrial mini-robot in cylindrical coordinates.

The transverse displacement (support) module (not shown on the diagram) is a welding station with guides on which a carriage moves. The remaining modules of the industrial mini-robot are mounted on it. The transverse displacement module accomplishes the work of industrial mini-robots in rectangular coordinates.

Each module has pneumatic distributors and valves, and also sensors to obtain information about the sequence of the work.

The enumerated modules were used in making up the planned PMR-0.5-254PV, PMR-0.5-200K, PMR-0.5-254KS and PMR-0.2-126PV robots, the main parameters of which are presented in the table.

Parameter	Robot Model			
	PMR-0.5-254PV	PMR-0.2-126PV	PMR-0.5-200K	PMR-0.5-254KS
Number of degrees of mobility	3		2	
Amount of horizontal displacements, in mm	254	126	200	Longitudinal 254 Transverse 330
Discreteness, in mm	2		-	Longitudinal 2 Transverse 4, 5
Amount of vertical displacements, in mm	50/35	30	35	
Positioning error, in mm	Positional-cyclic		Cyclic	Positional-cyclic
Number of connections with external equipment	Up to 10	Up to 5	Up to 12	Up to 10
Mass of displaced load, in kg	0.5	0.2	0.5	
Dimensions, in mm	670 x 340 x 710	350 x 220 x 400	810 x 220 x 600	805 x 705 x 500
Planned cost, in rubles	8,500	8,000	3,500	6,700

On the basis of the work of the PMR-0.5-254KS working in rectangular coordinates a complex was created for packing blanks of permanent magnets of complex form into cassettes. Oriented blanks arrive over a two-channel vibrating chute to a stop mounted on a support. The robot clasps two blanks with adjustable tongs, transports them and sets them on the basic elements of the cassette, shifting the coordinates of those elements accordingly (along the cassette thanks to the support and transversely, on account of a combination of multi- and two-range cylinders). Cassettes are automatically replaced in the complex.

The application of industrial mini-robots with discrete pneumatic drives made it possible to use a relatively simple system of programmed position control which is simultaneously a system for control of the entire complex for packing blanks in cassettes.

An example of the use of industrial mini-robots containing a two-range module with a vertically moving wrist is a robotized stamping complex intended for automatic feeding to a press, model KD-2124 with interchangeable punches, of previously punched out charts of four varieties of parts of the amplifier chassis and for the

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

the forced extraction of finished parts from the punch. The dimensions of blanks in the direction of feeding the punch are 30-110 mm and across the direction of feeding are 100-200 mm, the blanks are 0.5-1.5 mm thick and have a mass of 60-120 g and the material is ferromagnetic.

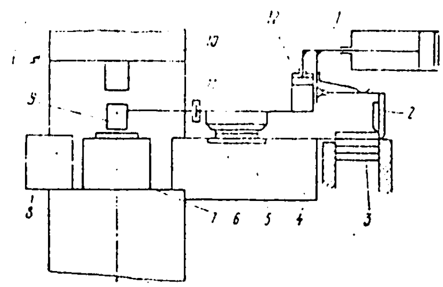


Figure 2. Pneumokinematic diagram of a robotized stamping complex.

The complex operates in the following manner (Fig 3). During movement of the robot rod forward the servo hook 2 separates the blank from the stack 3 and moves it into the intermediate position on the loader 4 in template 5 for final coordination of the blank. The electromagnet 6 then attracts the blank fed to the loader in the preceding cycle and sets it in the punch 7 of the press 10. Simultaneously the extensible electromagnetic clasp 2 set on hinge 11 withdraws from the punch the part bent in the preceding cycle and drops it into container 8. Pneumatic cylinder 12 is intended for raising and lowering the wrist containing the electromagnet and clasp during the withdrawal and insertion of the blank in the template of the loader and punch.

A kinematic elevator-accumulator with a step drive, developed in the Design and Technological Bureau, is used to feed the stack of blanks under the hook.

The use of three different working organs on the wrist of the robot in combination with the loader permits extracting previously oriented blanks from the stack, definitively coordinating the blank and setting it in the punch, extracting the finished part by force from the punch and dropping it in the container simultaneously after a single movement of the robot arm forward, which increases the productivity and precision of work of the industrial mini-robot as compared with ordinary two-handed rotating robots, and also permits dispensing with a special valve device to feed the part into the seizure zone.

According to preliminary data the working cycle of the robotized stamping complex must not exceed 3-3.5 days.

In the system of programmed control of a robotized stamping complex, provision is made for the gathering of information by means of external sensors and blocking in emergency situations. The main functions of the sensors are observation of the upper level of the stack of blanks in the cassette; monitoring the left and right positions of the robot arm and the upper and lower positions of the wrist of the robot hand, the correctness of placement of the blank in the punch, the removal of the part from the punch and the upper and lower positions of the press slide. The

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

logical circuit in the control system adds the sensor signals also when the robot mechanisms and the blank are in the correct position, and the removal of the article from the punch also gives permission for carrying out the working operation of bending. A second logical circuit disconnects the control system and stops the robot arm outside the stamping zone in the case of duplication of the press.

In the two other robotized stamping complexes of similar design, intended for secondary bending operations of small (40 x 34 x 0.3 mm) blanks, the electromagnetic clasps on the wrists were replaced by tongs and pincers. The blanks have small mass and are made of high-carbon steel. The use of electromagnetic clasps is inadvisable for such blanks because of residual magnetization and their possible "re-attachment" to the electromagnetic clasp which, in turn, affects the precision of blank placement in the template of the punch.

A distinctive feature of such a robotized stamping complex is the fact that in them provision is made, not for dropping the part in a container after stamping, but for re-setting it from the first into a second punch on the same press without loss of its oriented position for second bending and subsequent blasting of the finished part from the second punch with compressed air.

Robotized complexes, the work of which has been examined in the article, confirm the effectiveness of using robots of the industrial mini-robot series for the automation of technological processes.

2174
CSO: 8144/1449

UDC 621.7.077:62-85

MANIPULATOR WITH PNEUMOACOUSTIC SENSOR OF EXTERNAL INFORMATION

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 24-25

[Article by A. N. Shel'pyakov, engineer, and G. P. Isupov, candidate of technical sciences]

[Text] The automation of a number of technological processes, such as the transportation of parts from one conveyer to another when arbitrarily arranged, the painting of parts on a conveyer (the switching of the paint sprayer on during passage of the part through the working zone), the welding of non-rigid structures on a conveyer, etc, can be done only with manipulators which have sensors of external information. In spite of the variety of such technological operations, this is common to them: the need to search for a part when it is imprecisely placed on the working position, with subsequent withdrawal of the manipulator hand into the working zone.

Figure 1 shows a drawing of a very simple manipulator with a pneumoacoustic sensor of external information (a pneumoacoustic locator) which permits withdrawal of the manipulator hand into the zone in which the part is located regardless of the position of that part (within the range of action of the manipulator hand). The manipulator contains a pneumatic drive 10 on the rod of which is a pneumoacoustic sensor, consisting of a pneumoacoustic generator 6 of the type of a Hartman generator working at a frequency of 40 kHz, and an acoustopneumatic receiver 4, made to utilize the effect of destruction of a laminar jet by ultrasonic waves [1]. The pressure change on the outlet of the acoustopneumatic receiver 4 in the presence or absence of a part in the zone of action of the sensor reaches 2 kPa. The output signal from the acoustopneumatic receiver 4 is fed through the ST-55 "Volga" fluid-jet element to PF67-21 amplifiers 1 and 2, connected to the left and right cavities of pneumatic cylinder 10. The guide 7 together with the printing element 8 permits recording the displacement of the rod on the recorder 9 and determining the precision of positioning of the drive at different velocities of its motion and different distances from the sensor to the object 5.

The manipulator works as follows. At the initial moment the piston with the sensor is in the far left position. The sensor of external information is switched into operation by feeding the feeding pressures p_{01} and p_{02} to the generator 6 and receiver 4. In the absence of a part there are no reflected ultrasonic waves and pressure on the acoustopneumatic receiver 4 is maximal. It switches on the fluid-jet element 3, as a result of which the amplifier 1 operates and the piston with the sensor moves to the right toward the object 5. When the sensor reaches the

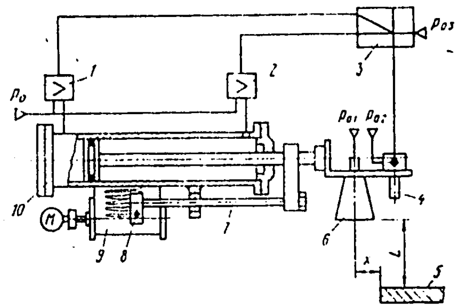


Figure 1. Diagram of manipulator with pneumoacoustic sensor of external information:
p_{03} -- feeding pressure of fluid-jet element;
X -- coordinate of stoppage points;
L -- distance between object and pneumoacoustic sensor.

object 5 the acoustic vibrations irradiated by the generator 6 are reflected from the object and impinge in the receiver 4, which leads to turbulization of the jet in the latter and a pressure drop on its outlet. The fluid-jet element 3 is switched on, the amplifier 1 is switched off and amplifier 2 operates, as a result of which the rod motion is decelerated and it moves to the left.

Upon emergence of the edge of object 5 beyond the limits of the zone of sensitivity of the sensor, the pressure on the outlet of the receiver 4 increases again, and this causes switching of the liquid-jet element 3 on, and the cycle is repeated. Thus the drive accomplishes scanning motion around the edge of the object: during movement of the object the drive overtakes it and again tracks the edge.

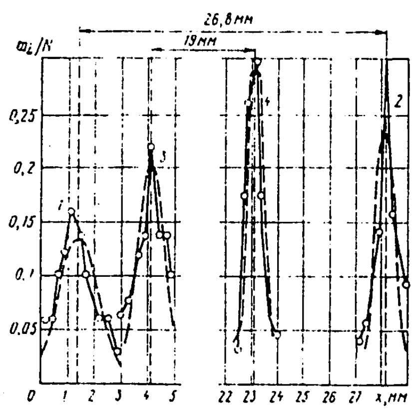


Figure 2. Distribution of coordinates of stoppage points of the drive of a manipulator equipped with a pneumoacoustic sensor of external information at L = = 200 mm:
1 & 3; at drive velocities toward the object (to the right) of 0.18 and 0.13 m/s respectively;
2 & 4: at drive velocities from the object (to the left) of 0.15 and 0.08 m/s respectively;
m_i: absolute frequency of coordinates of stoppage points in the definite interval X;
N: total number of measurements.

Statistical processing of the recording of scanning motion permits determining the precision of positioning of the drive. Shown on Figure 2 by a solid line are experimental curves of the distribution of coordinates of stoppage points of the drive at the distance L between the pneumoacoustic sensor and the object of 200 mm. Also given there is the amplitude of the scanning motion. The experimental data are well described by Gaussian curves (broken lines), constructed by smoothing out the empirical distribution [2]. The statistical processing was done on the results of 50 measurements (N = 50) at different drive velocities. The velocity was changed by installing a valve in the pneumatic line between the amplifier 1 and the cavity of pneumatic cylinder 10 (see Fig 1). On Fig 2 curves 1 and 2 were constructed in the absence of a valve and curves 3 and 4 with a valve 0.7 mm in diameter. In the former case the amplitude of the scanning motion was 26.8 mm, and in the latter it was 19 mm. The difference of drive velocities to the left and right is caused by the difference in effective areas of the pneumatic cylinder piston on the left and right sides. The table presents summary data on the precision of positioning as a function of the drive velocity v, where $\bar{X}$ is the coordinate of the center of grouping of the stoppage points; S is the mean-square deviation of positioning.

	Positioning error							
	Without valve				With valve			
v in m/s	0.18		0.15		0.13		0.08	
L in mm	100	200	100	200	100	200	100	200
Curve No (Fig 2)	-	1	-	2	-	3	-	4
$\bar{X}$ in mm	0.97	1.32	26.7	28.1	4.5	4	23.6	23
S in mm	0.42	0.73	0.27	0.43	0.39	0.5	0.24	0.33
2S in mm	0.84	1.47	0.54	0.86	0.78	1	0.48	0.67

The relatively low drive velocity of a manipulator with a sensor of external information does not limit its use. At a large velocity of the manipulator hand a circuit with deceleration is possible after a signal has been received from the sensor (circuits to accomplish that task are not shown on Fig 1). Therefore searching movements can be made at ordinary drive velocities, and the rate of tracking the edge of a part is selected as a function of the required precision.

The precision of positioning of the manipulator drive is determined by the forces of inertia of the mobile part of the drive, the time constants of individual elements of the system, the hysteresis of the liquid-jet element and the curvature characteristic $p_{out} = f(x)$--the dependence of the pressure p_{out} on the acousto-pneumatic receiver on the displacement x. The curvature of the characteristic is affected, other conditions being equal, by the form of the body to be located and the distance L (larger for plates and smaller for cylinders). The positioning error is reduced with decrease of the distance L, which is caused by increase of the curvature of the characteristic $p_{out} = f(x)$.

The maximum distance L_{max} at which the edge of the object is still traced depends on the directional pattern and the feeding pressure of the sensor. In addition, it is important to also know the influence of extraneous objects on the tracing precision. By selecting the feeding pressures p_{01} and p_{02} of the generator and receiver, a distance L_{max} of 300 mm was selected. It has been computed that extraneous objects at a distance of 500 mm or more from the sensor have no influence on the precision of tracing on the edge of the object.

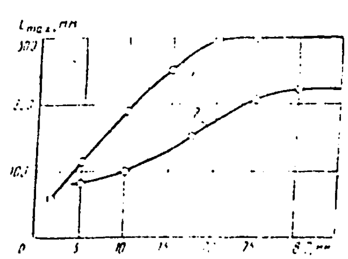


Figure 3. Maximum distance L_{max} as a function of the dimensions of the located surface. 1 & 2: for plane and cylindrical objects respectively.

Shown on Fig 3 are the dependences of the distance L_{max} on the dimensions (B = width, D = diameter) of plane and cylindrical bodies for the following feeding pressures of the circuit p_{01} = 0.2 MPa and p_{01}, p_{02} and p_{03} are 30, 10 and 7 kPa respectively. Plane objects are traced at a distance L_{max} of up to 300 mm, and cylindrical at a distance of up to 220 mm. The distance L_{max} can be increased by increasing the power of the pneumoacoustic generator, which is achieved, in turn, by increasing the feeding pressure p_{01}. Noise immunity of the sensor from external sound sources is assured by selecting sensitivity of the acoustopneumatic receiver only toward a definite frequency range [1] and a directivity of reception of acoustic vibrationa. These measures are accomplished by selecting geometric dimensions of the acoustopneumatic receiver and execution of its input in the form of a tube of a certain length and diameter.

A necessary condition of reliable work of a manipulator is careful soundproofing of the acoustopneumatic receiver on the side of the pneumatic generator, as during the interaction of direct and reflected waves interference is observed which leads to false operations.

Experimental investigations have shown that a tracing system (a pneumatic drive with a pneumoacoustic sensor of external information) permits remote tracing of an object of different form with an error of ±1.5 mm and withdrawal of the manipulator hand into the zone where the part is located when the latter has a flexible disposition on the working area. In that case the part can be both immobile and mobile with a velocity not exceeding that of the manipulator drive.

BIBLIOGRAPHY

1. Shel'pyakov, A. N., and Isupov, G. P. "Experimental Investigation of the Transition of Laminar Flow Into Turbulent Under the Effect of Acoustic Vibrations." INZHENERNO-FIZICHESKIY ZHURNAL, 1978, Vol 35, No 3.

2. RTM-44-62. Metodika statisticheskoy obrabotki empiricheskikh dannykh (Procedure for Statistical Processing of Empirical Data).

2174
CSO: 8144/1448

UDC 621.317.729.2:62-229.72

PROBE MANIPULATORS FOR INVESTIGATION OF SEMICONDUCTOR MATERIALS AND STRUCTURES ON PLATES

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 26-27

/Article by engineers B. N. Rachkin, E. G. Gasilov, B. I. Kudagin, and A. M. Gorbacheva/

/Text/ The need to improve and further develop the technology of production of semiconductor materials and structures on plates poses the paramount task of increasing the effectiveness of laboratory and industrial monitoring of their quality. In that case an especially important role is attributed to monitoring methods which have high metrological indicators.

To monitor those parameters it is necessary to assure a reliable electrical connection between the object being monitored or investigated and the measuring apparatus. Existing probe installations do not guarantee reliability of the parameters being monitored, are inconvenient to operate and, as a rule, are made by the users themselves under one diameter of the bases being monitored. Series-produced equipment of the type of "Zond A4" and "Zond A5M", used to monitor the parameters of integrated microcircuits and monitor semiconductor materials, and also structures on plates, prove to be economically disadvantageous, and their use to monitor electrophysical parameters by existing one-, two-, three- and four-probe methods presents considerable technical difficulty, and at times is completely impossible without major modifications of the design.

At the Penza branch of the All-Union Scientific Research Technological Institute of Instrument Making three modifications of probe manipulators have been developed which can be used to monitor the charge carrier concentration in epitaxial layers, the surface potential, the density of the surface state, the size and stability of the charge in the dielectric and on the interface, etc, by the method of volt-Faraday characteristics. Probe manipulators assure precise installation of probes on the base to be monitored thanks to the application of micrometer screws, which permit making a precise displacement of probes in all three planes, constant force of the probe needle spring on the base, monitoring of bases of different diameter, low capacitance in view of reduction of the metallic mass of the needle-holder and the use of high-frequency connectors and high-frequency cable which eliminate adjustments.

The manipulator design makes it possible to use them in both laboratory investigations and in the composition of measuring equipment in the process of debugging and monitoring of individual parameters of technological processes of production of semiconductor instruments and microcircuits.

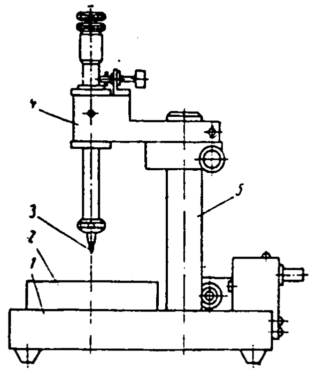


Figure 1. Single-probe manipulator
1 -- base; 2 -- stand; other key numbers explained in text

Figure 1 shows a single-probe manipulator. The displacement mechanism 4 assures a smooth rise and fall of the probe tip 3 on a base according to a given law, eliminates the probability of shock of the probe tip against the base and guarantees smooth rise of the contact pressure to the prescribed amount. Thus the base is protected against damage and the probe tip against bluntness.

The manipulator assembly includes a set of removable tables 2, intended for the arrangement and fastening of bases with diameters of 45, 60, 75 and 90 mm. The bases are fastened with a vacuum suction device. The measurement signal is taken and issued by an RK 50-2-22 cable and an SR 50-73F high-frequency connector, which permits assuring a stray capacitance of not more than 10 pF.

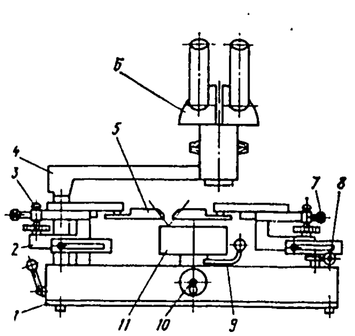


Figure 2

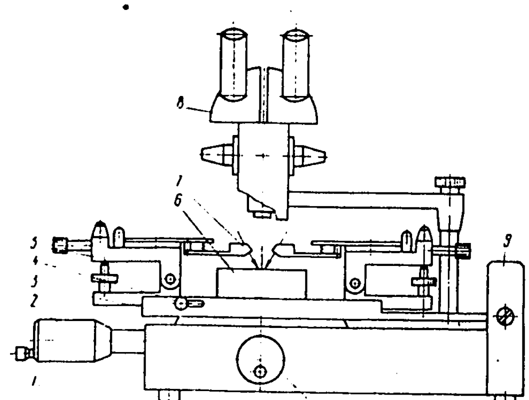


Figure 3

Figure 2 presents a schematic diagram of a two-probe manipulator; its technical characteristics are given in the table. It consists of a base 1 on which are arranged two movable probe-holders 2 with probe tips 5, a replaceable single-coordinate table 11 and a stand 4 with a microscope 6.

Parameter	Manipulator		
	single-probe	two-probe	universal
Diameter of bases to be monitored, mm		45.6-75.9	
Number of probes	1	2	1-8
Probe contact pressure, Ns	0.6	Controlled 0.05-0.6	
Displacement of object stage, mm			
on X coordinate		-	25
on Y coordinate		±25	
on Z coordinate	-	3	-
Displacement of probe-holder with probe, mm			
on X coordinate		50	
on Y coordinate		-	
on Z coordinate	17		15
Additional angular rotation of probe-holder, angular degrees	Within limits of object stage	15	
Fastening of bases on object stage		By means of vacuum	
Dimensions, mm	215x220x215	490x460x235	600x510x235

The removable stands permit arranging bases with diameters of 45, 60, 75 and 90 mm on them and fastening them with a vacuum suction device. The handle 9 of the stand, with the base of a cam-screw mechanism fastened on it, assures smooth rise and fall of the stand 11 by 3 mm. By means of the handle and micrometer screw 10 the stand with the base is moved in the range of ±25 mm with an error of positioning not worse than ±10 micrometers.

The probe tips are firmly fastened and have the possibility of independent movement on the X and Y axes, which is important when the sharp probe is precisely installed on a structure or contact area. For the installation of probes at a selected point of a monitored base it is necessary to move the probe-holder with handles 7 and 8 for the necessary distance along the X coordinate, and then with handle 9 raise the stand to contact the needle of the probe with the base. The moment of contact is set upon ignition of signal bulb 3.

Just as in the single-probe manipulator the use of RK 50-2-22 cable and the SR 50-73F high-frequency connector greatly reduces the stray capacitance arising in the measurement circuit.

The universal manipulator presented on Figure 3 permits solving a wider range of measurement tasks and implementing various methods of measuring semiconductor materials, epitaxial layers and structures, for example, one-, two-, three-, four- and five-probe methods. The universal manipulator is mounted on a base 1 where a replaceable two-coordinate stand 6, intended for the arrangement, fastening and positioning of microcircuit bases with diameters of 45, 60, 75 and 90 mm. Plates are also fastened by a vacuum suction device. By means of handles 2 and 10 of the micrometer screws precise movement of the stand along each of the X and Y coordinates by ±25 mm from the middle position is possible. The positioning error is not more than ±10 micrometers.

On stand 9 is a ring 3 which can be raised or lowered by 15 mm by means of handle 4. In recesses of the ring as many as eight modified probe-holders 5 with probe tips 7 can be arranged. Designwise they are similar to probe-holders with probe tips of a two-probe manipulator and have the same possibilities.

By means of a binocular microscope 8 mounted on a separate stand it is possible to not only monitor the quality of contact but also to establish the required distance between probes, for example, in monitoring the resistivity of the microcircuit bases.

Probe manipulators of all types include tungsten probe tips, but the installations permit using probes or tips also of other types, for example, nickel, depending on the measurement tasks.

Probe manipulators of the described types can be connected to measuring apparatus of any type without modification and are suitable for the solution of a very broad spectrum of tasks of measurement and monitoring.

2174
CSO: 1863/210

NEW MODELS OF YeS COMPUTERS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 30-31

/Article by A. B. Shirokova, methods specialist of the "Computer" pavilion, Exhibition of Achievements of the USSR National Economy/

/Text/ "New Models of YeS Computers"--such is the name of an exhibit which has been opened in the "Computer" pavilion, Exhibition of Achievements of the USSR National Economy. Two operating computers of the YeS "Ryad-2" are being demonstrated there now: the YeS-1035 and the YeS-1045. The YeS "Ryad-2" is a further development and improvement of the YeS "Ryad-1" computers. It represents a family of program-compatible third-generation computers with a unified software system and a single list of peripherals and data processing equipment /1-3/. The YeS-1035 and YeS-1045 computers are intended to solve a wide range of scientific-technical economic, information, logical and special tasks in an autonomous mode of operations and in a mode of systems information processing. They have program and information compatibility with the YeS computers produced earlier.

The YeS-1035 computer is a stationary universal computer which replaced the YeS-1022 computer, with an operating speed of 140,000-160,000 operations per second. The capacity of its main memory is 256K bytes and it occupies an area of 110 m^2. In comparison with the YeS-1022 the YeS-1035 has been provided with additional hardware and software which enhance its possibilities: a rechargeable microprogram memory, a diagnostic system, means of increasing reliability, high precision of computations with a floating point, and compatibility with the "Minsk-32" computer.

To provide rechargeability of the microprogram memory, the computer has an integrated circuit control memory capable of storing microprograms charged into the memory from the console panel magnetic tape store. If necessary the user can replace microprograms stored in the memory. This equipment assures simplicity and convenience of change of the computer configuration. By means of a special set of diagnostic microprograms charged, if necessary, from the panel store, defects are automatically located and indicated. The microdiagnosis system checks the working capacity of the processor for 10-15 minutes. Reliability is enhanced in the following manner. During reference to the main memory or the microgram memory a correcting code is formed which serves for checking the presence of error. Upon the appearance of a single error the information is corrected, and a double error only is recorded.

Provision also is made for automatic repetition of instructions during malfunctions. During the appearance of a malfunction in the processor, control is transmitted to the repetition microprogram, which restores the situation preceding the malfunction, and transmits control to the section of the microprogram where the malfunction occurred. Thus upon the appearance of random malfunctions normal functioning of the processors is assured. The error is registered in the memory for analysis. High precision of execution of computations with a floating point is realized by an instruction system which includes arithmetic instructions which process operands with a floating point with a length of 128 digits.

On the basis of application of software and hardware for simulation of "Minsk-32" computer programs, the conversion of programs into source languages and data transfer the compatibility of the YeS-1035 computer with the "Minsk-32" computer is assured, and this permits using the extensive fund of applied programs developed for the "Minsk-32" computers, thus freeing the user from costly reprogramming.

The central unit of the YeS-1035 includes a central processor and channels constructed on the basis of a single standard rack (YeS-2435). The main set is composed of a central processor with byte-multiplex and two selector channels, a main memory, processor power equipment, an operator's panel, a panel magnetic tape store and peripherals (magnetic tape stores with a control, interchangeable magnetic disk stores with a control, punched-card and punched-tape input/output devices, data preparation devices). In the unit demonstrated at the exhibit new magnetic disk stores with a capacity of 100 Mbytes (YeS-5066) were displayed for the first time.

The YeS-1035 computer software contains a disk operating system DOS/YeS, an operating system OS/YeS, software for the realization of compatibility with the "Minsk-32" computer and a set of technical servicing programs.

The YeS-1045 computer is included in models of YeS second-line computers. Its capacity for solving scientific-technical and economic tasks is 870,000 and 530,000 instructions per second respectively. The capacity of its main memory is 1-2 Mbytes. As regards design, the processor, input/output channels, main memory and power system are arranged in three standard stands: the processor and channels in the first, the main memory with a capacity of 1 Mbyte in the second, and the power system the channel-channel adapters and the logic retranslator in the third.

A number of new possibilities are provided for in the YeS-1045 computer: virtual memory facilities, an expanded set of instructions (as many as 183), high-speed channels and a flexible structure. The virtual memory facilities assure the creation of an expanded simulated main memory with the use of interchangeable magnetic disk stores. The virtual memory permits the programmer to use as many as 16 Mbytes of addressable memory in his program and eliminates the need to designate fixed regions of the real main memory for the program, and the operating system distributes the memory dynamically.

The high-speed channels (byte- and unit-multiplex) assure connection of a broad set of peripherals for various functional purposes to the computer. The carrying capacity of the input/output channels is 5 Mbytes/s. The flexible structure permits creating various configurations according to the requirements of the user.

A matrix processor, means of direct control and organization of a two-processor complex, channel-channel adapters and a logic retranslator can be used in the YeS-1045 computer. Application of a processor connected to the YeS-1045 computer.

across a special interface increases the effectiveness of the solution of problems in image recognition, the processing of geophysical data and other specific tasks. By creating a common field of the main and external memory for the two processors working under the control of one operating system, two-processor systems are constructed. Multimachine complexes are organized by means of channel-channel adapters, means of direct control and the general field of the external memory. An operating system assures the work of each computer in a complex.

Depending on the requirements of the user the functional and operating possibilities of the computer can be expanded through the introduction of additional hardware.

The YeS-1045 computer software has a powerful operating system OS YeS and the following sets of programs: technical servicing, non-autonomous tests of equipment and microdiagnostic programs.

On the basis of the exhibit "New Models of YeS Computers" in the "Computer" pavilion specialists can obtain qualified consultations on hardware and software of YeS computers. Consultations are provided by staff members of the Scientific Production Association "Algorithm" (Moscow) and the "Computer" pavilion of the Exhibition of Achievements of the USSR National Economy. One can become acquainted at the pavilion with the 1981 schedule for the holding of consultations on YeS computer hardware and software.

BIBLIOGRAPHY

1. YeS-1035--elektronnaya vychislitel'naya mashina (The YeS-1035 Computer). Moscow, Vneshtorgizdat, 1980.

2. Elektronnaya vybhislitel'naya mashina YeS-1045 (The YeS-1045 Computer). Moscow, 1979.

3. Katalog YeS EVM (YeS Computer Catalog). Moscow, Stroyizdat, 1980.

2174
CSO: 1863/210

DATA TELEPROCESSING SYSTEM HARDWARE COMBINATION

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 7, Jul 81 pp 3-4

[Article by V.I. Khomyakov, candidate of technical sciences, and Yu.M. Omel'yanchuk and V.P. Kozyr', engineers: "Features of the Design and Application of Data Teleprocessing Equipment"]

[Excerpts] One possible variant of a KTS [hardware combination] for small ISS's [information reference systems] and IUS's [information control systems] has been developed at the Scientific Research Institute of Peripheral Equipment (Kiev) [4].

Key Technical Characteristics of Hardware Combination

Interface with computer	Multiplex channel
Carrying capacity in bytes/s	20 K
Number of channels (together with US [interface device])	Up to 32
Type of communication channels which can be connected	Two- or four-wire assigned non-multiplex telephone and telegraph communications channels and physical circuits
Character code in transmission through channels:	
Telegraph	MTK-2M
Telephone	KOI-7
Transmission range through assigned telephone lines in km, respectively for four- and two-wire connection systems with transmission speeds in bits/s of:	
200 and 600	25-20 and 10
1200 and 2400	16-12 and 8-6
4800 and 9600	10-7 and 5-4
Transmission range through assigned telegraph communication lines with transmission speeds of 50 and 100 bits/s in km	15-8
Interface with terminal equipment	S2 (V24) interface
Method:	
of data exchange with terminals	Semiduplex
of establishing communication	Automatic

of data transmission	Sequential asynchronous
of modulation in transmission of data through telegraph channels	Uni- and bipolar direct-current signals (40 to 25 mA)
of modulation in transmission of data through telephone channels	Bipolar direct-current low-level signals ($\pm$ 0.76 mA)
Combination of number of telephone and telegraph channels in UUPD [data transmission control device] and US-1 [interface device] units	Variant
Size of picture on screen of video terminal in mm	250 X 190
Number of:	
Characters in line	54 (64)
Lines	20
Size of character in mm	5.5 X 3.5
Character set	96
Formation of display	On television raster
Regeneration frequency in Hz	50
Capacity of buffer storage in bytes	1280
Input and editing facilities	Keyboard and tag

In order to produce hard copy of the information read out on the screen, the ability to connect a printer to the BU [buffer unit] of the VT-1 video terminal is provided. The hardware of the STD [data teleprocessing system] can be expanded by adding to it asynchronous modems (e.g., YeS-8006's), used for considerably increasing the number of remote video terminals, as well as by connecting type DZM-180 high-speed printers which serve the purpose of printing out data from the video terminal's screen.

8831
CSO: 1863/216

UDC (621.39 6.001:681.322):681.327.00

ASPECTS OF THE EMPLOYMENT OF PERIPHERALS IN AUTOMATED DESIGN OF INTEGRATED CIRCUITS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 7, Jul 81 pp 6-7

[Article by V.G. Tabarnyy, candidate of technical sciences, and V.A. Mironchenko and M.I. Lobak, engineers]

[Text] The development of microelectronic technology has confronted developers of integrated circuits (IS's) with a number of problems associated with the combined miniaturization of electronic components and computer hardware (SVT) units. The employment in this equipment of made-to-order integrated circuits which have considerably improved the technical and economic indicators of SVT has necessitated the creation of comparatively inexpensive but efficient automated design systems (SAPR's) for IS's (SAPRIS's). This is due to the fact that microcircuits made to order are often produced in relatively small lots, as the result of which the time required for and the cost of designing them are decisive factors.

As testified to by know-how gained in the development of SAPRIS's in our country and abroad, these requirements are satisfied most completely by SAPR's functioning in the man-computer dialog mode. Such systems have been given the name interactive (ISAPRIS's).

Usually the most formalized stages in the design of integrated circuits are performed automatically in an ISAPRIS, such as checking and diagnosis of input data, various transformations, numerical modeling of electrical processes, production of control data media and the like. In the dialog (interactive) mode problems are solved which have not been formalized at the present time. For example, at the structural design stage the problem arises of placing topological figures on a plane surface with the possibility of deforming them for purposes of reducing the area of the integrated circuit chip. This problem can be solved effectively only by the combined work of a human being--the developer of the integrated circuit--and a computer, whereby the designer is assigned the creative role in this dialog and the computer makes a constant check of the permissibility of his actions and makes possible the required transformations of numerous data.

This article elucidates know-how gained in the development of the first phase of an ISAPRIS for integrated circuits employing MIS [metal-insulator semiconductor] transistors, which makes it possible to create integrated circuits to the fourth degree of integration for microelectronic elements and computer hardware units.

The ISAPRIS which has been developed represents a complicated multiprocessor and multifunctional dialog system of the human-computer type having in its structure the following pieces of equipment: computing complexes (M-4030 computer, SM-4 UVK [control computer complex]) with a basic combination of peripherals; special-purpose peripheral and terminal units (a KPA-1200 coordinatograph, EM-709 graphic information coders, an A5122 graphic information readout and preparation unit (USPGI), AP7251 and AP7252 graph plotters, A5433 EPP graphic displays and a wide-format A54310 graphic screen-type designer's console (an ShG EPP-2)); interfaces, connection equipment and office facilities; ASVT 2.2 [modular hardware system] OSRV [real-time operating system] and DOS [disk operating system] operating systems; control, service and applied programs of the system; and organizational and methodological support systems.

The technical characteristics of the peripheral and terminal equipment in the above list determine many characteristics of the system as a whole. For example, one important characteristic is the requirement of the necessary coordinate precision with good response of the system in the process of dialog between the designer and computer. With regard to response the peripheral equipment can be divided arbitrarily into equipment with comparatively poor response (coders, graph plotters and coordinatographs) and equipment with relatively good response (graphic displays--the designer's screen consoles). This has resulted in division into three basic steps of the process of dialog designing at various stages of the development of integrated circuits:

Preparatory, at which large arrays of data (functional and electrical circuits, sketches of the topology of integrated circuits) are entered by means of graphic data readers and the data are processed in the computer together with a check and diagnosis of errors (fig 1).

The dialog per se, when the editing of texts and drawings takes place, along with the input of new relatively small fragments and manipulations with groups of graphic items; these operations are performed by means of the graphic display, which is the integrated circuit designer's basic tool at his automated work place (ARM).

The output of design results onto graph plotters and coordinatographs for the final documentation or a check and the preparation of new modifications at individual stages of the development of integrated circuits (fig 2).

At each stage the requirements for these peripherals are determined by such morphological variables of representations of the integrated circuit being designed as the functional diagram, sketch of the topology, electrical circuit diagram and topology.

The semiautomatic graphic information coders used in the ISAPRIS which has been developed make it possible to code points, lines, outlines and characters of various types with a sufficient degree of coordinate precision and with acceptable speed. When working with the USPGI the designer does not have to be concerned with the horizontal placement of a drawing on the base of the plotting board, since "shrinkage" of the material on which the circuit is represented is automatically taken into account in the USPGI. The existence of a direct link between the USPGI and an SM-4 UVK makes it possible for the developer to carry on an active dialog with the computer already at the stage of the entry of graphic data, which considerably shortens the time for the correction of errors at the labor intensive stage of entering and checking source data.

Figure 1. An Integrated Circuit Designer's Automated Work Place; Stage of Coding Large-Scale Integrated Circuits with an EM-709 Coder

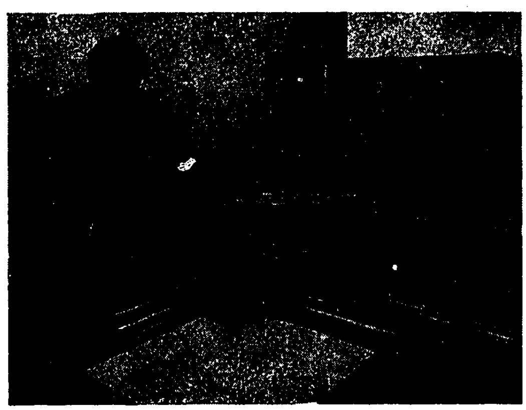


Figure 2. Stage of Drawing Large-Scale Integrated Circuits on a KPA-1200 Coordinatograph

The basic means of enabling a dialog mode in the ISAPRIS described are graphic displays--designer's screen consoles, the A5433 EPP and A54310 ShG EPP-2. The presence of a graphic display possessing broad capabilities in the representation and editing of graphic and alphanumeric information gives the interactive mode considerable flexibility.

At the disposal of the integrated circuit designer are three screen console operating modes--text, mosaic and vector--each of which provides convenience in the designer's work at a specific stage in the development of a microcircuit. Line segments and an extensive set of characters can be placed on the display's screen in two simulated layers and are isolated by means of intensified image brightness, which makes it possible easily to distinguish individual sections of a circuit. Capabilities for the selective formation or erasure of individual sections of an image, shift and rotation of fragments of a circuit ensure an effective and efficient editing mode.

The units for reading out graphic information in the ISAPRIS are, in addition to the display, an AP7251 roll-type graph plotter and a KPA-1200 coordinatograph. The AP7251 serves the purpose of displaying functional and electrical circuits or fragments of them and various graphs, i.e., is used when high coordinate precision is not necessary. The KPA-1200 coordinatograph, which possesses sufficiently good accuracy with relatively slow response, is used when reading out the integrated topology of an integrated circuit.

In the ISAPRIS under consideration the interactive mode to one extent or another affects all stages of the design of integrated circuits and in the end result determines the serviceability and vitality of the system as a whole.

Data arrays entered into the computer and obtained by means of it in the dialog mode are constantly checked. Diagnosis is carried out in various aspects according to the maximum reasonable number of criteria.

The arrangement, scale, type, form and nature of the graphic and symbolic data read out to the peripherals used in the ISAPRIS make possible visibility, simplicity in reading, completeness of representation, familiarity of form to the user, and the possibility of direct reading without additional processing.

In the dialog mode the user can select the data processing mode, inhibit or permit the execution of specific procedures while entering the appropriate instructions by means of the display's control keys and select optical tags or working instructions. Furthermore, the system's internal supervisory routine automatically adjusts for the operating mode specified, executes the appropriate routines and selects the required peripherals.

In the ISAPRIS the computer hardware and peripheral equipment are arranged according to the hierarchical principle: A basic M-4030 computer is used at the top level for solving problems of great dimensionality, e.g., functional and electrical analysis of integrated circuits, of the arrangement and preliminary tracing of elements on a chip and the like. An A5433 EPP graphic display is offered to the developer for the M-4030 computer for the purpose of dialog. A small SM-4 computer provides the designer's group with several automated work places (up to four), having in their composition an A54310 ShG EPP wide-format graphic designer's screen console and an A5122 USPGI. Communication between the SM-4 and M-4030 computers is carried out at the level of the transfer of data recorded on magnetic tape.

The peripherals included in the structure of the ISAPRIS have hardware for functioning in the packet processing mode with time sharing (with dialog and without dialog in the design process) and the independent mode, which gives the system considerable flexibility.

The ISAPRIS which has been created is a system of the open-end type and makes it possible to add both new programs and new peripherals. The comprehensive approach used in developing the ISAPRIS and the level of software and hardware have made it possible to create a relatively inexpensive system enabling the flawless design of integrated circuits containing up to 10^4 MIS transistors. The use of this system guarantees, with considerable shortening of the time required for designing integrated circuits, the development of technical solutions which are not inferior in quality to those produced by the nonautomated (manual) method.

8831
CSO: 1863/216

UDC 681.325.5-181.48

PROGRAMMER FOR MICROPROCESSOR DEVICES

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 7, Jul 81 pp 23, 26

[Article by A.M. Gladkov and N.V. Nagayets, engineers, and V.A. Teslenko, candidate of technical sciences]

[Text] The choice of program storage and its loading are included among the major problems facing developers of microprocessor devices and systems. Reprogrammable storage units (PPZU's) at the present time have found extensive application as program storage units for microprocessor devices. For example, in the creation of any microprocessor device, even with the presence of ready software and a formally debugged program, the problem arises of checking the good working order and correctness of the functioning of the microprocessor section together with other sections of the device. At this stage it is feasible to use a PPZU as the program storage. In the creation of small-lot and unique devices and systems it is not economically advantageous to use as a program storage programmable ROM masks--in this case also PPZU's are appropriate. The use of them in microprocessor devices and systems increases their flexibility and makes it possible for the user himself to modify both the operating algorithm of equipment and the exchange of information between units.

The entry of data into a PPZU can be accomplished either with a computer provided with the appropriate interfaces, or by means of special units--programmers (loaders), which must be provided with the appropriate keyboard and display making it possible to change information in the PPZU and to check its entry.*

A programmer developed by the authors is discussed in this article. Its structural diagram is presented in fig 1.

The programmer is designed for working with devices or systems executed on the basis of a microprocessor (MP) of the K580IK80 type (or having a similar instruction word length). The basis of the programmer is an electrically reprogrammable storage unit possessing the capability of automatic erasure, entry and readout, and a RAM of the static type. A tape reader and a patch panel located on the front

*Lukichev, N.I., Nayfel'd, V.L. and Rog, G.V. "Reprogrammable Semipermanent Storage," PRIBORY I SISTEMY UPRAVLENIYA, No 11, 1979.

panel of the programmer serve the purpose of entering data into the storage. Data line and address line display units are provided for the purpose of checking the data entered. An address assignment unit assigns the address of a number.

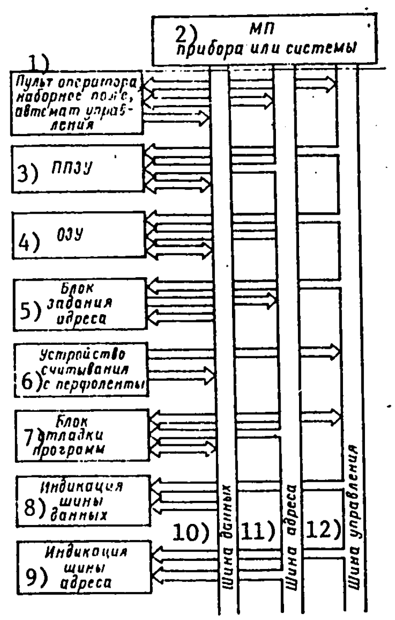


Figure 1. Structural Diagram of Programmer

Key:

1. Operator's console, patch panel, automatic control unit
2. Microprocessor of device or system
3. PPZU
4. RAM
5. Address assignment unit
6. Tape reader
7. Program debugging unit
8. Data line display
9. Address line display
10. Data line
11. Address line
12. Control line

Communication between individual units of the programmer and other devices is accomplished through a data line for data, an address line for addresses and a control line. The programmer has several operating modes: entry of data into the programmer; transcription of data within the programmer; operation of the programmer as a program storage with microprocessor devices or systems; transcription of data from the programmer into the storage (PPZU) of microprocessor devices and systems; program debugging mode; check of the operation of the microprocessor of microprocessor devices and systems.

The entry of data into the PPZU can be accomplished either directly or via the RAM. In the latter case an important disadvantage of a PPZU is eliminated, involving the loss of data in the entire array of the storage when it is necessary to edit (erase) the contents of a single storage location.

Data enter the RAM or PPZU from an eight-bit data line. In turn data can enter the data line from the patch panel located on the front panel of the programmer, from a tape reader, from a microprocessor (in the check mode), a RAM and a PPZU.

The capability of transcribing data from a PPZU into the RAM is also provided in this unit.

Data entering the data line are displayed on the data line display in octal or decimal code. The display mode is selected by the operator. The entry of data can be accomplished either optionally, for any address, or sequentially, beginning with a specific prespecified address. The address line is 16-bit. The code of the initial address is entered byte by byte from the data line by the operator.

A built-in program debugging unit makes possible the step-by-step running of programs with the connection of a microprocessor device. The step-by-step debugging of programs can be performed from any prespecified address, since a mode for halting a program at a specific step is provided. The address at which it is necessary to halt the running of a program is entered in the programmer's memory. When this address agrees with the current address the programmer generates a "Stop" instruction and transfers the microprocessor into the mode of the passing of instructions according to machine or instruction cycles. A display of the address and state of the data line takes place with each halt of the microprocessor.

It is possible to check the contents of any storage location in the program debugging mode. When an error is detected the entire array (256 lines) of the storage in which the error is detected is transcribed from the PPZU into the RAM. After elimination of the error the array is again transcribed into the PPZU.

An important disadvantage of electrically reprogrammable storage units and, consequently, of microprocessor devices, too, designed on the basis of PPZU's, is the limited data storage time (3000 to 10,000 hours) with the power cut off, as well as the gradual erasing of information in the readout process. For the purpose of eliminating this disadvantage it has been suggested that these microprocessor devices be provided with a special memory regeneration subroutine. Regeneration consists in sensing by the microprocessor of the contents of a specific storage location, of switching the PPZU over to the entry mode, and of "thoroughly recharging" the same storage location. By using a microcomputer's RAM it is possible to carry out this procedure for specific arrays of the storage.

A flow chart of the regeneration routine (23-byte capacity) of the K580IK80 microprocessor is shown in fig 2. It functions as follows. The initial address is entered in register pair HL, and the final address of the storage field to be regenerated into pair BC. The microprocessor reads into register A the contents of the storage location whose address, M , is indicated by register pair HL. After this follows an extended cycle for the entry of the contents of register A into the storage according to the same address. With this a readin pulse 10 ms long is formed by means of hardware. Then follows the byte-by-byte subtraction of the contents of pair HL from the contents of pair BC. If the result of the subtraction equals zero, then this means that regeneration of the entire specified storage field has been accomplished and the routine has finished its job. With a non-zero result the contents of pair of registers HL are increased and a return to the next read cycle takes place--entry according to label M3 .

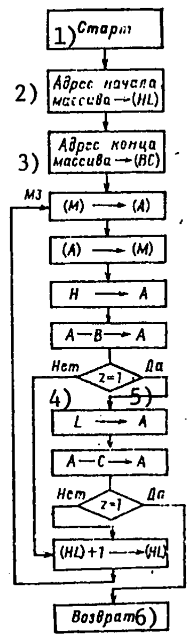


Figure 2. Flowchart of Regeneration Routine: H, B, L and C represent general-purpose registers of the microprocessor and Z is a bit of the condition register

Key:

1. Start
2. Address of beginning of array
3. Address of end of array
4. No
5. Yes
6. Return

Regeneration of the PPZU according to this subroutine can be performed both on the instruction of the operator and automatically each time the microprocessor device is turned on. This subroutine can be included among test subroutines for checking the working order of the microprocessor device.

Key Technical Characteristics of Programmer Developed

Capacity of memory in bytes:	
RAM	4 K
PPZU	4 K (with the possibility of increase to 16 K)

Readin time in microseconds:	
RAM	0.5 to 1
PPZU	$(10 \text{ to } 20)\cdot 10^3$
Readout time in microseconds:	
RAM	0.5 to 1
PPZU	1.2 to 1.5

The programmer comes in a unified frame with overall dimensions of 480 X 480 X 240 mm.

8831
CSO: 1863/216

MULTIMACHINE AND MULTIPROCESSOR SYSTEMS

Moscow NOVOYE V ZHIZNI, NAUKE, TEKHNIKE: SERIYA "RADIOELEKTRONIKA I SVYAZ'": EVOLYUTSIYA VYCHISLITEL'NYKH SISTEM in Russian No 3, Mar 81 (signed to press 12 Feb 81) pp 42-54

[Chapter 6, "Multimachine and Multiprocessor Systems," from the monthly "Radioelectronics and Communications" serial, "What's New in Life, Science and Technology": "The Evolution of Computer Systems," by Professor Yevgeniy Pavlovich Balashov, doctor of engineering science, author of 12 books and 96 inventions, specializing in computer hardware, and Arkadiy Petrovich Chastikov, docent, candidate of engineering science, author of over 40 scientific articles and many inventions, Izdatel'stvo "Znaniye", 38,260 copies, 64 pages]

[Text] In tracing the history of computer hardware development we can observe that efforts to enhance the capacity and reliability of computer systems proceeded in two directions: first, improvement of hardware, primarily by increasing speed of operation, and second, realization of new architectural ideas and principles of organization of parallel computation.

Progressive miniaturization based upon semiconductor electronics approached its physical limits when limitations associated with electronic lithography became operative. This finds expression in the fact that voltages cannot be reduced in proportion to the dimensions involved; the electrical fields in integrated circuits therefore increase with a reduction in linear dimensions. Closely related to hardware miniaturization is the question of density, that is, the number of components or circuits per unit of single-crystal area, while power dissipation has proved the most serious limitation upon achievement of high density.

Taking this into account, the second direction now holds out the greatest promise for success in achieving the desired objective.

The development of multimachine and multiprocessor computer systems has been one of the dramatic examples of the appearance of new architectural solutions and principles of parallelism. Three principles underlie the development of these systems: parallel execution of a large number of operations, variability in structure and uniformity of design.

The idea of multimachine and multiprocessor systems was first mentioned in the works of the Soviet scientists E. V. Yevreinov and Yu. G. Kosarev and in those of D. Slotnik, although the Soviet priority must be pointed out here, since the first work by E. V. Yevreinov and Yu. G. Kosarev, "O vozmozhnosti postroyeniya vychislitel'nykh sistem vysokoy proizvoditel'nosti" [The Possibility of Building High-Capacity Computer Systems], published by the Siberian Department of the Academy of Sciences of the USSR in 1962, appear some six months before the publication of D. Slotnik's work on the SOLOMON project, while the technical realization of these ideas was accomplished in our country six years before (the Minsk-222 homogenous general-purpose computer).

The idea of creating multimachine and multiprocessor computer systems was further developed in the works of the Soviet scientists V. M. Glushkov, Yu. S. Golubev-Novozhilov, A. V. Kalyayev, M. A. Kartsev, G. P. Lopato, B. N. Naumov, V. V. Przhilkovskiy, I. V. Prangishvili, V. S. Seminikhin, Ya. A. Khetagurov, V. G. Khoroshevskiy et al., as well as in those of the foreign scientists D. Kholland, M. Leman, L. Koktsely, G. Vang, M. Flin [transliterated] et al.

A multimachine computer system consists of several computers, each of which functions under the control of its own operating system and has means of exchanging information with other machines. Multimachine computer systems found their fullest expression in the development of homogenous computer systems, the ideas of which were formulated in 1962 in the Institute of Mathematics of the Siberian Department of the Academy of Sciences of the USSR.

The term "homogenous computer system" is understood to refer to an aggregate of computers functionally interacting with one another via a regular program-adjusted communications network.

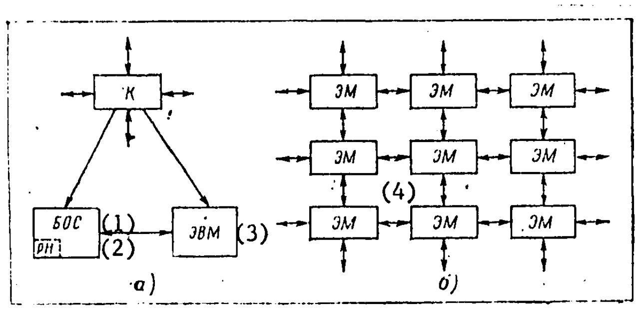


Figure 6: 1 - system operations unit; 2 - adjusting register; 3 - computer, 4 - primary machines.

Figure 6*a* is a diagram of an homogenous computer system. A switching unit (K), the computer (3) and the system operations unit (1) comprise the primary machine (4). The channels and switching units form the communications network between machines. The switching unit generates a set of connecting functions permitting establishment of communication between a given machine and its immediate neighbors. The adjustment register determines the type of connecting function, that is, by changing the contents of the adjustment register it is possible to predetermine any switching between input and output switching unit terminals. Figure 6*b* shows the topological layout of primary machines in an homogenous computer system.

Work on the planning and design of the first homogenous computer system, the Minsk-222, was begun in 1965 by personnel of the SO AN SSSR [Siberian Department of the Academy of Sciences of the USSR] Institute of Mathematics together with the design bureau of the Minsk Works imeni G. K. Ordzhonikidze under the direction of G. P. Lopato, State Prize winner and corresponding memebr of the Academy of Sciences of the USSR, and V. V. Przhiyalkovskiy, State Prize winner and works director, the first model being turned out in April 1966. The Minsk-222 is an homogenous ring computer system with a partially modifiable structure and two-way channels of communication between the primary machines.

The number of computers in the system may vary from 1 to 16. Minsk-2 or Minsk-22 computers are employed in the primary machines.

The basic advantage offered by the Minsk-222 homogenous computer system was demonstrated in the fact that with the large capacity of its internal memory as compared with the Minsk-22 and the high speed of its communication channels it substantially reduced the time required to solve complex problems--by some 5-10-fold, 20-fold for some problems with a system having three primary machines.

Work on subsequent homogenous computer system designs built further on the experience accumulated in designing and operating the Minsk-222. The following year, 1967, saw the development of an homogenous real-time linear control system. It was designed to control technical scientific experiments and production processes.

One of the promising directions in the development of multimachine computer systems would then become the construction of these systems on a minicomputer base.

In 1976, working in conjunction with the Severodonetsk Impul's Scientific Production Association, personnel of the Institute of Mathematics, SO AN SSSR, engineered the design for the MINIMAKS program-switched minicomputer system. The MINIMAKS is an homogeneous system with program-adjusted channels of communication between primary machines. Each primary machine comprises a computer complex (M-6000 or M-7000) performing an information-processing function and a system unit interfacing between primary machines. It is interesting to note that the ratio of the cost of the system unit and that of the MINIMAKS processor does not exceed 0.5, while this ratio for multiprocessor systems with a common bus with the RDR-11 computer processor is 1.2.

The MINIMAKS system may function autonomously, as part of high-capacity concentrated computer complexes or as part of distributed computer networks, it lending itself to successful utilization for both solving scientific and economic problems and process control.

Among later developments of homogeneous computer systems we should mention a minicomputer system (SUMMA) created in 1977 on the basis of the domestically manufactured Electronika 100 and Elektronika 100I minicomputers, a system oriented toward operating as part of automatic systems controlling industrial production processes. The Institute of Mathematics, SO AN SSSR, and NITsEVT [Scientific Research Center for Electronic Computer Equipment] have now engineered the design for a general-purpose homogeneous computer system based upon the YeS-1060 computer. Its overall speed is determined by the number of YeS-2060 processors and by ratings may range from 2 to 20 million operations per second.

The development and manufacture in our country of homogeneous computer systems is an important achievement in the sphere of building multimachine computers with programmable structure.

There are many similarities between multimachine and multiprocessor systems, since they both serve the same basic purpose--simultaneous execution of operations in the system, while distinctions, frequently not entirely clear-cut, are based upon frequent use of the term "multiprocessor processing" in cases in which it is not properly identified. There is, however, an important difference between them, a difference based upon the degree and limit of partitioning: a multimachine system comprises a number of individual computers (even if there may be direct communication between them), while a multiprocessor system is most frequently a special-purpose computer with multiprocessor units. It should be pointed out in this connection that it is simpler to construct multimachine systems from series-manufactured computers with standard operating systems than to construct multiprocessor systems requiring the solution of certain difficulties connected with realization of the full extent and development of a special operating system.

Strange as it may seem, the roots of the development of the principles of multiprocessor processing are clearly to be sought in the depths of analog computing technology and the principles of analog analog computation. An analog computer is in fact an example of a multiprocessor system, in which there occurs parallel processing (solution) of tasks (systems of differential equations) in all operating units of the structure. So there we have them, new turns in the dialectic spiral.

Beginning with the first ENIAK, digital computers have incorporated the ideas of multiprocessing. The ENIAK's arithmetic unit had 20 adders, which permitted the simultaneous execution of several addition and subtraction operations.

A modular design with identical processors was for the first time incorporated in the Burroughs D-825 system, which was manufactured in 1962 for military application. Total system storage, which comprised 16 modules, was divided between all four processors. The creation of one of the first advanced operating systems, the ASOP (automatic operating and scheduling program), was an important feature of the system. Work on the development of various types of multiprocessor systems was subsequently successfully undertaken in many countries.

Let us now move ahead to a discussion of the principles of the organization of multiprocessor systems, but first let us define more fully the present-day concept of this type of computer. The American dictionary, "National Standards for Information Processing," defines a multiprocessor system as "a computer incorporating two or more data-processing units with integrated control." But this definition is incomplete. The requirement that a multiprocessor system have integrated control is without doubt extremely important since it must have an integrated one-operation system; but this definition does not encompass the concepts of partitioning and interaction, which constitute the essence of multiprocessor processing methods. A multiprocessor system must be able to partition internal memory among all processors and i/o units with all combinations of memory and processor. Figure 7 shows the basis of multiprocessor organization. An important feature of the interconnection is the level at which it occurs. The level at which interconnection (interaction) is permitted must be more flexible in multiprocessor systems than in multimachine systems, where a data aggregate normally consitutes the unit of interaction.

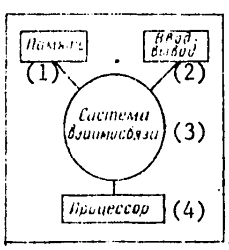


Figure 7

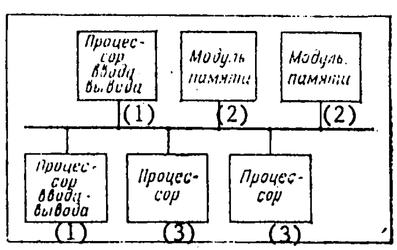


Figure 8

Key: Figure 7: 1 - memory; 2 - i/o; 3 - interactive system; 4 - processor. Figure 8: 1 - i/o processor; 2 - memory module; 3 - processor.

It is precisely the combination of the amplified concepts of partitioning and interaction at all levels that fully characterizes the hardware and software required for a multiprocessor system, which may now be defined as a system incorporating two or more processors of approximately equal capacity and which have access to full storage and i/o channels, the system as a whole being controlled by a single operating system providing interaction between processors and programs at the level of jobs, tasks, data aggregates or data elements.

The system of interaction and such important factors as topology and the functioning of the central part of the system provide the key to classification of multiprocessor systems.

F. Enslow and a number of other authors have identified three very distinct organizations employed in multiprocessor systems: systems with a common bus, those with matrix switching and systems with common storage.

The common bus system is a version of the simplest interaction having a common channel of communication linking all functional units (Figure 8). The common bus system is employed in the fabrication of small, simple multiprocessor systems, simple in the sense that the interactive subsystem may consist of no more than a multiwire cable.

The advantages this system offers include the following: low hardware cost, a simple interactive subsystem and easy modification of hardware structure; its drawbacks lie in the facts that overall system capacity is limited by bus transmission speed, bus failure is accompanied by major system malfunctions and that enlargement of the system by the addition of other units may reduce overall capacity.

Figure 9 illustrates the organizational principle of systems with matrix switching (matrix organization). System storage is organized such that each of its modules may send or receive information from any processor or i/o unit via any bus running from the processors and i/o units.

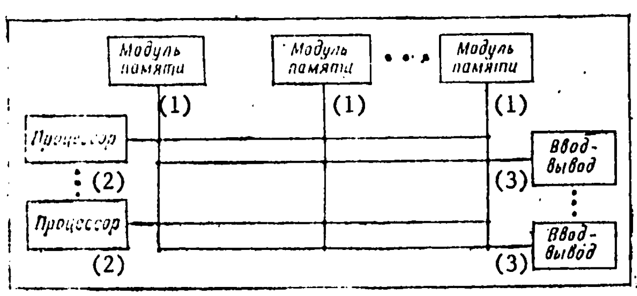


Figure 9. Key: 1 - storage module; 2 - processor; 3 - i/o.

The address of a word transmitted via a bus is decoded at each matrix node to determine the storage module into which the word should be written or from which it should be read. The operation in any given storage module does not depend upon operations executed in neighboring modules.

One of the special features of this system organization consists in the fact that it incorporates the most complex interactive subsystem, although its functional units are simple and cheap (control and switching logic is located in the matrix nodes, decoders and switching units); enlargement of the system is theoretically limited only by the capacity of the switching matrix, which may be increased within the limits imposed by the initial structure; reliability of switching, as well as that of the system as a whole, is insured through segmenting and redundancy, which makes it easy to differentiate the system to eliminate malfunctions. This organization is, generally speaking, economical only for multiprocessor systems.

If control and switching logic and random priority selection lean toward storage modules, the result of multiprocessor organization will be a system with common storage (multichannel storage) (Figure 10). This organization theoretically fully corresponds to both single-processor and multiprocessor structures.

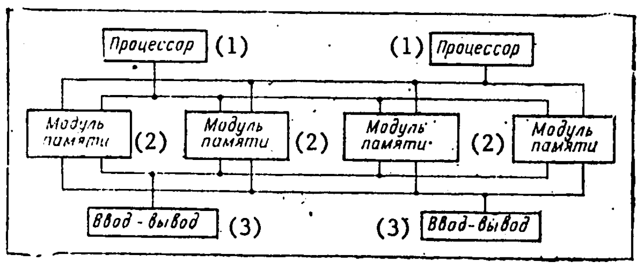


Figure 10. Key: 1 - processor; 2 - storage module; 3 - i/o.

Each storage access channel in this system is assigned permanent priorities insuring the access of any unit of the system to the various storage modules. All channels of the common memory are identical from both the structural and functional points of view. These channels conventionally take the form of a series of cable connections; to which channel an i/o or computing processor is connected is therefore of no importance; the priority of each system unit's access to storage is established by selection of the connector.

This system organization requires expensive storage modules, since many of the control and switching circuits are connected to these modules, as well as a large number of cables and connectors. Different modifications of the system are determined by the number and type of available storage access channels. The structure with common storage permits the achievement of high capacity levels throughout the system.

The development in the 1970's of high-capacity computer systems with speeds into the tens and hundreds of millions of operations per second represented a further elaboration of multiprocessing thinking directed toward increasing system capacity and the achievement of a high degree of parallelism.

High-capacity computer systems are broken down on the basis of architectural characteristics into SISD, MISD, SIMD and MIMD structures.

Let us now look at some characteristic examples of high-capacity computer systems: conveyor, or pipeline, systems, matrix, or vector, systems and associative systems.

The idea of the pipeline principle of control organization, which achieves a high degree of parallelism in the processing of command and data flows, belongs to Academician S. A. Lebedev (1964); he referred to it as the "water line" principle. He had still earlier (1957) put forward the idea of employing combined operations in a computer. In pipeline, or conveyor, computer systems, the processor consists of a conveyor (pipeline) of information processing units, in which the output data of the i-th processing unit become the input data for the i+1 unit etc. Storage modules store externally generated data as well as intermediate results transmitted from corresponding processing units.

The computation process is thus broken down into a number of steps, each of which has its corresponding processing units and storage modules, each processing unit receiving its own flow of commands. Because of the parallel operation of the units of the entire conveyor in "filling the pipeline," the end processing unit puts out the computation results at short intervals of time equal to the time required to execute the slowest step, although the theoretical total process time is equal to the total time required for processing by all units.

The STAR-100 system (STAR = STring and ARray), built in 1973, is a typical example of a pipeline computer system. The system's processor comprises three pipelines, each of which consists of as many as 30 information processing units. Two of the pipelines are for the execution of operations with a floating decimal point on data vector pairs, the third for processing data not organized in vectors. All processing units operate in parallel, each of them operating with either specific data vector elements or with scalar elements. System storage comprises up to 32 modules and stores programs and data. The STAR-100 system has a capacity of approximately 100 million operations per second. It should be pointed out that higher capacities have been achieved in later versions of computer systems incorporating the pipeline processing principle (200 million operations/s with the Cray-2, 300 million operations/s with the SUPER STAR).

E. V. Yevreinov and V. G. Khoroshevskiy point out that "pipeline computer systems represent the limit to the modification of serial computer architecture. Possibilities of increasing computer speed by means of pipeline information processing are therefore essentially limited: when results are transmitted via a conveyor, the sequence of operations cannot be of arbitrary length, if only because of the ramification of the computation process."

The matrix multiprocessor organization underwent further development in high-capacity matrix, or vector, computer systems. This system is a matrix of identical processors comprising an arithmetical logic unit and storage and which is controlled by a single flow of commands. Information may be exchanged between any processors in the matrix via a communications network. A single control unit exercises direct control of the matrix. Centralization of the control unit and a high degree of parallelism in information processing and storage permit substantial increases in system operating speed.

Uncompleted plans for the SOLOMON, the first matrix computer system, were published in 1962. A modified version of this system was technically realized in May 1972 in the ILLIAC-IV matrix computer system. This system has a speed of 200 million operations/s; it comprises $6 \cdot 10^6$ electronic components, occupies 930 m^2 and weighs 75 tons. The ILLIAC-IV matrix consists of 64 identical processors for executing operations over vectors. Its control unit consists of special-purpose PDP-10 and PDP-11 computers.

The processor may be in either a passive or an active state: in the former instance it is inhibited from executing commands from the control unit but permitted in the second.

The ILLIAC-IV system is employed to solve complex, large-dimension problems (systems of linear algebraic equations, linear programming, matrix arithmetic, rapid Fourier transforms etc.); since the system is controlled by a single command flow, its structure permits parallel execution of identical operations on 64 data sets written into processor memory.

The capacity of the ILLIAC-IV may be compared with that of high-capacity third-generation computers taking the example the solution of a linear programming problem with 10,000 variables and 4000 limits. A third-generation machine requires 6-8 h to solve this problem, the ILLIAC-IV less than 2 min.

A drawback to the matrix systems consists in the fact that in the passive state some processors may remain idle for long periods of time; it should also be pointed out that difficulties have arisen in the establishment of communication between processors; but despite this, the creation of matrix systems constitutes an important achievement in the development of computer system architecture.

Advances in integrated technology and the creation of large-scale integrated circuits have made possible the development of a computer system, differing somewhat from the matrix system and having an SIMD structure, referred to as an associative system. The essential component of this system is the associative memory, in which, as is known, words are located on the basis of their content. In an associative computer system a large number of linked processors comprise the associative memory. The creation of an associative memory is based upon utilization of logical principles. Each word contains enough logic to compare it with a known word. The associative processor employs this type of associative memory as internal memory, each word and its corresponding logic comprising a processing element.

In contrast with the Neumann architecture, comparison takes the place of summation in an associative processor and is performed by a process similar to scanning a table. Since the search operation (table-scanning procedure) is executed in parallel with all words in memory, the associative processor can sum with a high degree of parallelism.

The Staran-IV system, developed by Goodyear Aerospace in 1971 and having a speed of 40 million operations/s, was the first associative computer system. The system was modernized in 1972; it had a speed of 500 million operations/s and was being used in an airport air-traffic control operation. The Staran-S, Staran S-500 and Staran S-1500 associative computer systems have been developed or are being developed.

In 1971, Bell Telephone and System Development came out with an interesting version of an associative computer system referred to as PEPE (Parallel Element Processing Ensemble). It has a capacity of 300 million operations/s. This system has much in common with both matrix and associative systems; it is therefore sometimes referred to as an associative-matrix system, although it does have its own special features. For example, the system is capable of processing two independent flows of commands, while in contrast with the ILLIAC-IV system its basic processors do not interface directly with their immediate neighbors. The ultimate objective of the development of the PEPE system is the creation of a system comprising 288 basic processors, each with a capacity of 1-5 million operations/s, operating under the control of a CDC-7600 computer.

Associative computer systems, which are an efficient means of processing large files of information, have only a single control device, which decreases the reliability of the system and narrows its range of applications.

8963
CSO: 1863/203

YES 9004 KEY-TO-TAPE DATA ENTRY DEVICE

Riga AVTOMATIKA I VYCHISLITEL'NAYA TEKHNIKA in Russian No 3, May-Jun 81 p 100

[Advertisement for data entry device by exporter VTO "Isotimpex"]

[Text] The YeS 9004 device is intended for direct recording on magnetic tape of information entered by the keyboard. It provides the capability of finding a particular block of data to check it and correct the data on the tape when necessary.

The unit has a built-in buffer storage that holds a full block of data before recording it on tape. This allows immediate correction of errors in input noticed by the operator and elimination of them by backing up the storage address and entering the correct character.

The data on the magnetic tape is recorded in blocks with a length of 80 or 160 characters. The device has a keyboard interlock that operates during recording on tape and the read cycle after recording which allows the operator to maintain a constant rhythm in entering data from the keyboard.

During the write cycle, the contents of buffer storage are not disturbed--the data is preserved for checking which is effected with immediate read out after writing. Data read from tape is compared bit by bit with data in storage.

In the "checking" mode, data blocks intended for checking are read and input to storage one after the other. Each character of the block is re-entered using the keyboard from the source document. The code of the input character is automatically compared to the code in storage. When both codes are identical, the operator may continue checking.

In "retrieval" mode, a particular data block is automatically retrieved by comparing each block recorded on tape to the identifier previously entered in storage from the keyboard. Retrieval is executed at the rate of about 1,000 blocks/minute.

Each YeS 9004 can have a block-by-block display of the current data block.

At the customer's option, the YeS 9004 performs the following operations: copying of data from one device to another, data printout from magnetic tape to printer, and data entry from a card reader.

Specifications:

recording density	32 bits/mm
recording method	RZ [return to zero] - 1
tape speed	39.6 cm/s
buffer storage size	160 bytes
block length	80 and 160 bytes
record format	by 1 0
basic operating modes	data entry, checking and retrieval
other modes	program entry, program checking, printing, entry from card reader
programs	2 independent
display	block-by-block
power	220V ±10-15%

Exporter: VTO "Isotimpex", Sofia, ul. [street] Chapayeva, 51. Telephone: 7361, Telex: 022731, 022732.

8545
CSO: 1863/223

S5 MICROCOMPUTERS

UDC 681.32-181.48(47+57)

SOFTWARE AND BASIC APPLICATIONS OF 'ELEKTRONIKA S5' MICROCOMPUTERS

Moscow MIKRO-EVM "ELEKTRONIKA S5" I IKH PRIMENENIYE in Russian 1980
(signed to press 3 Nov 80) pp 84-153

[Part of chapter 4, chapter 5 and conclusion from book "'Elektronika S5' Microcomputers and Their Application", by Mark Petrovich Gal'perin, Vladimir Yakovlevich Kuznetsov, Yuriy Aleksandrovich Maslenikov, Vladimir Yefimovich Pankin, Viktor Panteleymonovich Tsvetov and Aleksandr Ivanovich Borovskoy, Izdatel'stvo "Sovetskoye radio", 34,000 copies, 160 pages. This third installment completes the translation of this book. The first two installments appeared in the USSR REPORT: CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY, JPRS L/9703 (pp 100-108) and L/9766 (pp 1-73)]

[Text] 4.3 Resident Facilities for Automation of Programming

Two large groups of microcomputer users can be identified. They differ by how they use standard microcomputer software. The first group includes users who use microcomputers for series and mass produced articles with one type of program in ROM (read-only) memory (this group of users willingly and readily uses cross-SARP [facilities for automating program development], and is interested in the development of these means); the second group includes users who use microcomputers for small-scale control systems, laboratory research, teaching purposes, and control of sets of equipment whose configuration changes quite frequently, in other words, that class of applications where the use of permanent memory is possible only for various standard algorithms of the given class but the principal specific programs change frequently. This class of users requires the formulation of "at-hand" resident facilities that accelerate the process of formulating specific programs during frequent readjustment or during actual operation of the systems.

Resident SARP of microcomputers of the Elektronika S5 family have been developed and continue to be developed to meet the needs of this group of users (see Figure 4.8 below). A distinctive feature of the development of resident SARP is their full language compatibility with corresponding cross-facilities, which provides a simple transfer of raw texts from the technological computer to the microcomputer and in the other direction.

It should be observed that the choice of the first realization of a second-level language for resident facilities (as also for cross-facilities), the BASIC language, results from a number of factors, among which the following are singled out: convenience of program development and debugging in the interactive mode;

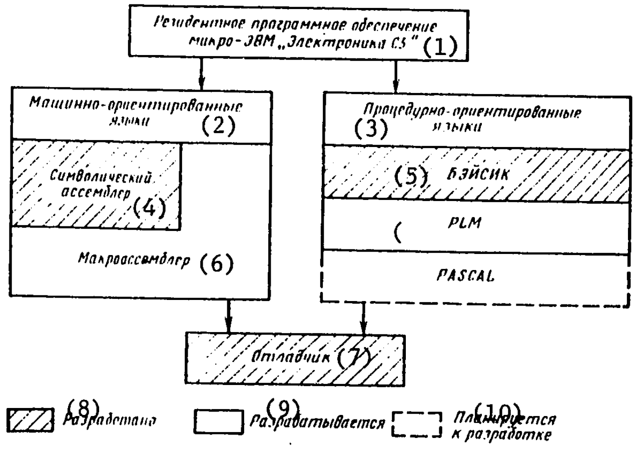


Figure 4.8. Resident Microcomputer SARP

Key: (1) Resident Software of the Elektronika S5 Microcomputer;
(2) Machine-Oriented Languages;
(3) Procedure-Oriented Languages;
(4) Symbol Assembler;
(5) BASIC:
(6) Macroassembler;
(7) Debugging Unit;
(8) Developed;
(9) Under Development;
(10) Planned for Development.

---

availability of a fund of ready-to-use programs (subroutines); the possibility of use in series-produced microcomputers and the widespread current use of BASIC. A version of this language that is analogous to the version used extensively in the powerful calculators of the Wang Labs Company, with well-chosen texts that accompany the dialog process and funds of algorithms used in the processes of control and data processing, was used for the Elektronika S5 microcomputer. A user of the Elektronika S5 microcomputer, equipped with a translator from BASIC, is able to work in the computation and programming modes and correct and debug programs directly from the console, using a keyboard and the input-output tape punch of a teletype.

An important feature of the translator from BASIC is the possibility of introducing blocks that insure compatibility between the specific software of the microcomputers of the Elektronika S5 family and models that are not included in this family. Moreover, a translator from the language built on the compilation principle makes it possible to prepare programs for single-board and single-chip microcomputers. This approach speeds up the programming of specific programs for control systems based on these kinds of microcomputer configurations and offers the possibility of efficient debugging of these programs within a simulation complex based on a multiboard microcomputer equipped with this translator.

The realization of BASIC language in the Elektronika S5-02 microcomputer relies on a hardware fixation of the software of the language in ROM LSIC's [large-scale integrated circuit], the development of several versions of the language calculated for work with a broad assortment of peripheral units and permitting enlargement of their number and interlinking with versions of dispatcher systems, and modification of algorithms of the language during the operation process and when building up the functional capabilities of the microcomputers [22].

To improve the effectiveness of resident program debugging, it is also useful to design and set up a resident debugging complex in the ROM LSIC of the Elektronika S5-02 microcomputer. The complex is made up of programs together with the peripheral units of the microcomputers (the Kvant video monitor display and a functional keyboard). The structure of the editing-debugging complex is shown in Figure 4.9 below. The editing-debugging complex provides the following services: program debugging (including debugged texts; formulation of input punched tapes for the SARP with texts prepared for realization in the ROM LSIC; representation and editing of data on the display screen).

The subsystem of editing-debugging complex programs that organize the process of debugging specific microcomputer programs offers the programmer broad opportunities to monitor performance of the program being debugged with extensive indicators of the elements that are monitored. Special software-hardware facilities greatly speed up debugging, reducing its interpretation costs from 15-20 commands for one command performed to three commands.

The subsystems "Dialog with the Display Corrector" and "Dialog with the Program Editor" edit original and debugged texts of programs. This process is controlled by the functional keyboard.

These facilities fully insure resident debugging of target programs at their own level and preparation of data for introduction to the system to automate the design of ROM LSIC's.

The punched tape version of the editing-debugging complex has the capability of parametric retuning for black-and-white and color displays with arbitrary screen size, various combinations of peripheral units, and changes in the volume of main memory [23]. Figure 4.10 below gives a structural diagram of the use of resident means of automating program development.

4.4. Software.

Basic principles. The extensive use of microcomputers in different sectors of the economy inevitably means that in addition to professional program developers of software systems based on preceding generations of computers, a broad range of nonprofessionals in this field of activity, with or without experience and/or education in the software field, will be involved in designing the software of microcomputer-based data control, processing, and transmission [DCPT] systems.

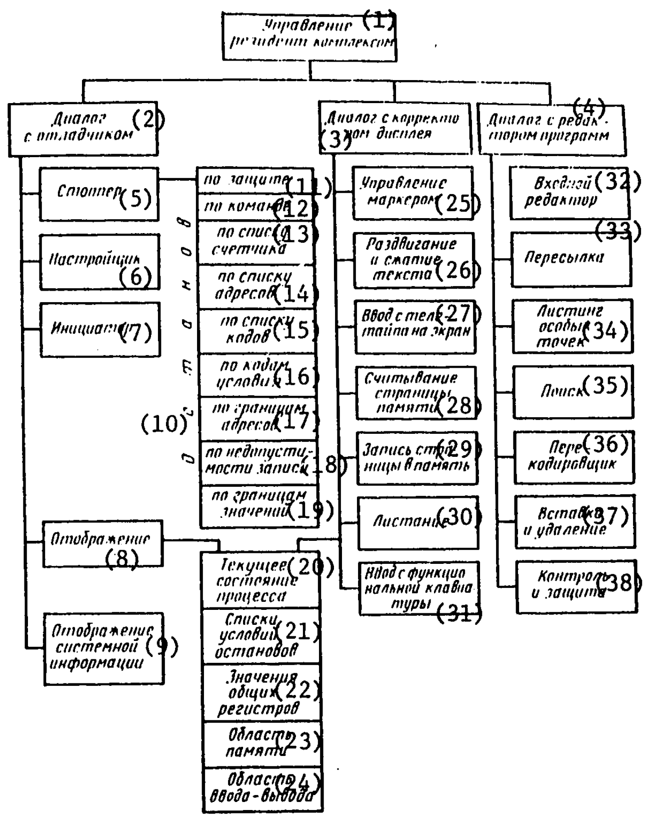


Figure 4.9. Resident Editing-Debugging Complex

Key:
(1) Control of Resident Complex
(2) Dialog with Debugging Unit;
(3) Dialog with Display Corrector
(4) Dial;
(5) Stopper;
(6) Tuner;
(7) Initiator;
(8) Representation;
(9) Representation of System Data:
(10) Stops for:
  (11) Security:
  (12) Command;
  (13) Counterlist;
  (14) Address List;
  (15) Code List;
  (16) Condition Codes;
  (17) Address Boundaries;
  (18) Impermissible Entry;
  (19) Boundaries of Values;
(20) Ongoing State of Process;
(21) Lists of Stop Conditions;
(22) Values of Common Registers;
(23) Memory Domain;
(24) Input-Output Domain;
(25) Marker Control;
(26) Text Expansion and Compression;
(27) Input to Screen from Teletype;
(28) Reading a Page of Memory;
(29) Writing a Page into Memory;
(30) Turning a Page;
(31) Input from Functional Keyboard;
(32) Input Editor;
(33) Recopying;
(34) Listing Special Points;
(35) Retrieval;
(36) Recoder;
(37) Insertion and Deletion;
(38) Monitoring and Security.

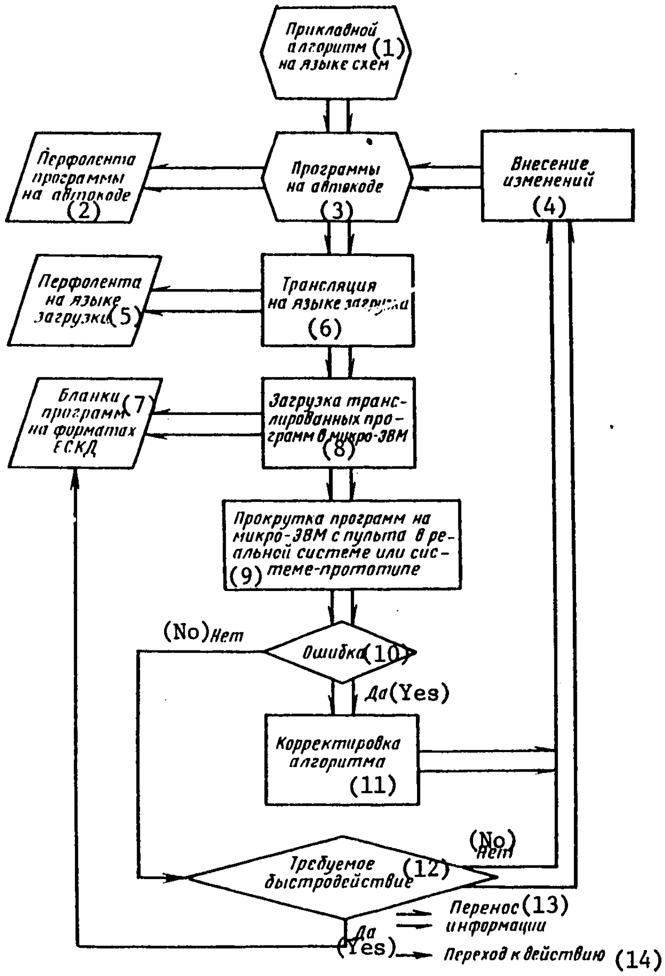


Figure 4.10. Development of Target Programs on the Microcomputer.

Key:
(1) Applied Algorithm in the Language of the Circuits;
(2) Punched Tape of Program in Autocode;
(3) Programs in Autocode;
(4) Insertion of Changes;
(5) Punched Tape in Loading Language;
(6) Translation to Loading Language;
(7) Blanks of Programs in YeSKD [Uniform System of Design Documentation] Formats;
(8) Loading Translated Programs in Microcomputer;
(9) Scanning Programs on the Microcomputer from the Console in the Real or a Prototype System;
(10) Error;
(11) Correction of Algorithm;
(12) Required Speed;
(13) Transfer of Data;
(14) Branch to Action.

At the same time, most of these systems work in real time, have more than one job, and interact with different pieces of peripheral equipment. This requires that these systems have special organization of control over problem-solving and exchange with peripheral units. But the development and debugging of such means is one of the very most difficult questions in the field of designing the software of control systems.

It follows from this that equipping series-produced models of microcomputers with facilities developed by specialists who have considerable working experience in the field of control technology and have mastered all the possibilities offered by the different models of the microcomputer family is an essential job for the enterprise that is developing microcomputers. This solution also reduces the labor-intensiveness of developing target software and improves the quality of its work through the use of standard software that has been thoroughly tested in a large number of applications.

There are two possible ways to realize this software: on punched tape (or magnetic tape) and in a permanent memory unit. Because the primary area of use of microcomputers (with respect to extent and national economic efficiency) is in the sphere of controlling equipment in the immediate vicinity of data sources, not in the sphere where tape and disk storage (application of minicomputers) are widely used, alternate feeding of such programs from punched tape complicates operations and reduces the reliability of the microcomputer-based hardware.

Therefore, the best way to achieve these facilities in microcomputers is by storing programs to control problem-solving and exchange with peripheral units in an ROM memory. Despite a certain "conservatism" in this solution and a certain redundancy in each variation of it for insuring the maximum capabilities within given constraints, this method successfully pays for itself through simplicity of content and use, and maximum reliability.

This approach was followed in designing the dispatcher systems of the Elektronika S5 microcomputers, which are delivered with particular versions assigned in the ROM LSIC. This makes them, for the programmer-consumer of specific software, inseparable from all the other hardware of the microcomputer. This is what gives us the right to call dispatcher systems program equipment.*

Two versions of dispatcher systems have been developed and are used for Elektronika S5 microcomputers: the teletype version and the modular version. Both were developed taking into account a number of general principles. A number of basic concepts were assumed in the dispatcher systems whose generalized structure is represented in Figure 4.11 below.

---

* The authors also think that any program means (dispatcher, automation of programming, and the like) delivered to the customer in the permanent memory of a microcomputer can be classed with this specific equipment.

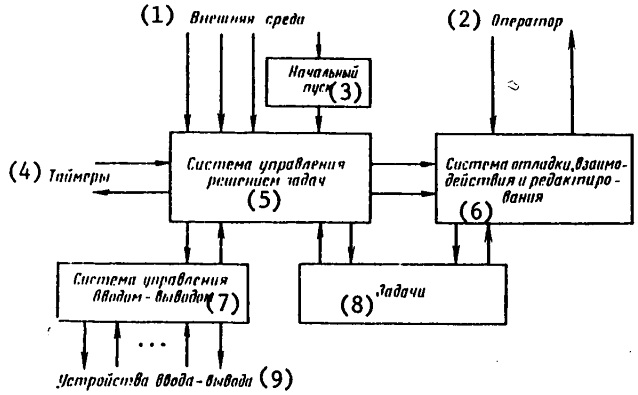


Figure 4.11. Structure of the Dispatcher System.

Key:
(1) External Environment;
(2) Operator;
(3) Initial Start;
(4) Timers;
(5) Problem-Solving Control System;
(6) Debugging, Interaction, and Editing System;
(7) Input-Output Control System;
(8) Problems;
(8) Input-Output Devices.

A problem is the work of a sequence of commands that determine the specific program. The sequence is initiated through a system of interrupts. The problem is the basic unit of specific software. A bit position in the RPPR [possibly register of program interrupts] corresponds to each problem (or group of problems). The bit positions must be arranged in conformity with the priority of the problems or groups of problems as determined by the algorithms of system work. The larger the number of the bit position in the RPPR, the higher the priority of the problem or group of problems will be. Certain problems are interrupted by others depending on a given security matrix. The initial security matrix, which is fed through the dispatcher system in the process of initial launching of the system, is triangular. Necessary corrections in it can be made by entering the necessary codes in the No 2 common registers of all groups.

The supervisor is the part of the dispatching system that provides the multiprogramming and real-time modes. The basic functions of the supervisor are: parallel performance of problems; performance of problems in conformity with established priorities; including problems on the initiative of hardware or a program; one-time or cyclical solutions; controlling the work of the timers; and, organizing queues for the peripheral units.

The macrocommand is an element of the language for interaction between problems and the supervisor. It requests subroutines that are included in the dispatching system. These subroutines are used to expand the language of the system of microcomputer commands. In this case it is necessary to perform certain system procedures. The problem receives the result of performance of the macrocommand in the form of a return code.

The macrosubroutine is a supervisor subroutine initiated by a macrocommand.

A macrocode is the initial parameters (alphanumeric information) that are inserted in the macrocommand for the work of the macrosubroutines. The standard part of a macrocode contains system information, while the functional part reflects the specific features of the different macrocommands.

In functional terms the dispatching system includes the supervisor, control of input-output, and facilities for interaction with the operator.

Control of input-output interacts directly with the peripheral units assigned by the problem. In this case the processes of input-output take place both simultaneously and in parallel with problem-solving.

Facilities for interaction with the operator make possible interactive communication with the system, offering certain debugging devices and editing outputted information.

The teletype version of the dispatching system (TVDS). The minimum configuration of a computing system based on microcomputers includes the microcomputer itself with a teletype adapter and the teletype itself, which provides the capability of feeding data from a keyboard or punched tape and outputting to a printer or punch. The TVDS was designed to control the work of this system.

The TVDS is delivered in the above-described configuration with an Elektronika S5-01 (02) microcomputer, and is also available with an Elektronika S5-12 microcomputer. The basic purpose of both configurations is to provide debugging complexes based on single-board microcomputers for processing equipment. At the same time, the specifications of this dispatcher system and the supporting hardware delivered in these configurations are also entirely adequate to serve as the basis for designing corresponding data control, processing, and transmission (DCPT) system.

The minimum configuration of the computer system based on microcomputers determined the volume of the TVDS program, which is 1,024 words (the minimum memory construction — two ROM LSIC's). These LSIC's are located at addresses 6800-67FE on the board of the Elektronika S5-01 (02) microcomputer (addresses 6800-77FE are free and may be used for target and standard programs). The LSIC's with monitoring and preventive microcomputer tests are located at addresses 7800-7FFE. Where the TVDS is used in the Elektronika S5-12 computer, one of the zones of 1,025 words is connected to the addresses of the TVDS (6000-67FE). The second zone of 1,024 words may be connected to the addresses 7800-7FFE and the tests may be arranged in it.

Problems are engaged following interrupt signals by means of the supervisor program START. The new problem is engaged by program by the microcommand REQUEST, and the work of each problem is ended by the microcommand OUTPUT. When the problems interact with the teletype the microcommands EXCHANGE and WAIT are used.

The TVDS includes all the above-listed components: the supervisor, the teletype driver (program to control interaction with the telegraph), and editor of text information outputted to the telegraph. Data exchange with the telegraph is organized by the telegraph driver through the macrocode EXCHANGE, and the dispatching system provides for several of the most widely used types of exchange (text, code, and numerical).

The facilities of the dispatching system insure the console mode of work of the TN [expansion unknown] (entering "1" or zeroing the bit positions in the "console" cells) and initiation of a problem from the telegraph keyboard. The editing facilities make it possible to print different tables by the principle of filling in "gaps" in the text headline.

The modular version of the dispatching system. This version is figured for microcomputers with a comparatively broad assortment of adapters for peripheral units, telegraphs, photo reader, tape punch, and EPM [expansion unknown]. Like the TVDS, the modular dispatching system (MDS) organizes problem-solving, services interrupts, controls timers, and interacts directly with the peripheral units ordered by the target problem. Other major distinctions between the MDS and the TVDS, besides the broad range of peripheral units serviced, are the availability of a program mechanism that makes it possible to include new program modules for communication with additional peripheral units in the MDS, and the expanded assortment of facilities for communication with the operator, including additional possibilities of program debugging.

The MDS is designed only for the appropriate configurations of Elektronika S5-02 microcomputers. In this case this machine can be used as a debugging complex for systems with single-board microcomputers and broad possibilities of using data input-output and preparation equipment and as the foundation for constructing systems to control data processing and transmission with a broad range of target programs that require elaborate facilities for dispatching (supervising) jobs occurring in real time.

The comparatively powerful configuration of a computer system based on microcomputers necessitated the use of a program level and a microprogram level in the MDS. On the program level the MDS occupies 3K words at addresses 6000-77FE, which means that they are allocated together with the monitoring-preventive tests on a single board of the permanent memory unit in the microcomputer. The microprograms of the MDS occupy 2K words of the microprogram permanent memory. Problems are engaged following interrupt signals by means of the START program. Program control of the state of a problem is accomplished by feeding the macrocodes REQUEST, ABORT, and EPISODE; work with the timers is determined by feeding the macrocode TIMER, which is used basically to trigger any timer. The problem interacts with peripheral units by means of the macrocodes EXCHANGE, INTERROGATION, and CANCEL. The problem can also use service macrocodes available in the MDS.

The MDS edits information that is being prepared for outputting to peripheral units, and organizing facilities for interaction with the operator makes it possible for the user himself to determine the necessary type of editing by means of an appropriate program written by him in the editor language. In addition, facilities for interaction with the operator enable a user to control the microcomputer in the language of dispatcher directives, and one of them (INSERT) offers the possibility of monitoring the course of performance of programs to print out the memory, the state of the interrupt system, and the contents of the general register during performance of these commands.

Autonomous exchange with peripheral units. The problem of minimizing hardware and debugging programs in the Elektronika S5-21 microcomputer is solved by putting programs for exchange with the teletype, photo reader, and punched tape and programs that organize the debugging regime through the teletype in the ROM LSIC. This approach makes it possible to construct a debugging complex directly on the basis of this machine and additional main memory modules without drawing on more elaborate models of the family. The programs for autonomous exchange insure full logical interaction with peripheral units through a digital input-output unit located in the microcomputer without additional MFM's [microprocessor functional modules]. These programs are also controlled (ON, OFF, TRANSFER, from regime to regime) through these digital input-output units. These programs also contain transfer subroutines (see Figure 4.12 below).

Control of the performance of programs is done by means of teletype. During this process it is possible to start and stop a program and order printing of the addresses of the commands being performed and the contents of the registers and assigned memory during performance of the particular command [9].

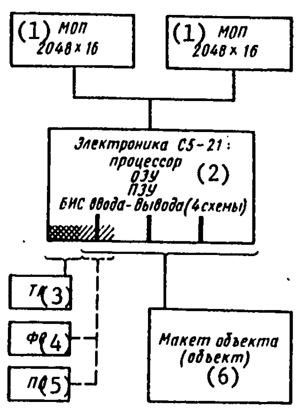


Figure 4.12. Minimum Debugging Complex Based on Elektronika S5-21 Microcomputer

Key:

(1)	Main Memory Module;	(3)	Teletype;
(2)	Elektronika S5-21 Processor Main Memory, Permanent Memory, Input-Output LSIC (Four Circuits);	(4)	Photo Reader;
		(5)	Tape Punch;
		(6)	Model of Object (Object).

4.5. Preparation of Information for the Fabrication of ROM LSIC's

The use of microcomputers in DCPT systems led to the use of ROM memory programmable by photo masters to realize the programs of the target algorithms. In addition, the convenience to the microcomputer user was significantly improved if the standard software of the operating systems, blocks of standard subroutines, blocks of debugging facilities, and translators from assembler languages and high-level languages were designed in the form of software in ROM memory modules, either permanent or replaceable depending on the type of software and/or the conditions of its use. This type of realization of software, which is not new in comparison with certain earlier machines such as the UM1NKh, Elektronika K-200, and others, acquires a number of special characteristics in the microcomputer that force one to somewhat modify the approach to this realization, concentrating greater attention on it than was given to ROM memory in earlier generations of control machines.

The impossibility of operational corrections of a semiconductor permanent memory that is programmable by photo masters makes it necessary, on the one hand, to raise the reliability level of debugging and, on the other hand, to absolutely preclude the subjective factor in the process of preparing debugged information or making ROM LSIC's, turning this process over entirely to the computer. The fact that the basic elements of the storage and processors are of the same type (see Table 4.2 below) shapes the general approach to designing both the logical structures of the microcomputer and its ROM data storage circuits, using in both cases the same system for machine design of LSIC's. Therefore, combining cross-facilities of automating program development with a system for machine design of

Table 4.2

Parameters of Comparison	Ferrite Permanent Memory	Semiconductor Permanent Memory
Possibility of Automated Receipt of Documents	Yes	Yes
Automated Manufacture	Yes	Yes
Possibility of Operational Correction after Fabrication	Yes	No
Common Basic Elements in Storage and Processor	No	Yes

LSIC's, in other words setting up an integrated system to design microcomputers [24], provides the information base for automatic transmission of information on target programs to the subsystems that design the photo masters for making ROM LSIC's (See Figure 4.13 below). The importance of this kind of integration is obvious for series production of single-chip microcomputers with special-purpose ROM memories.

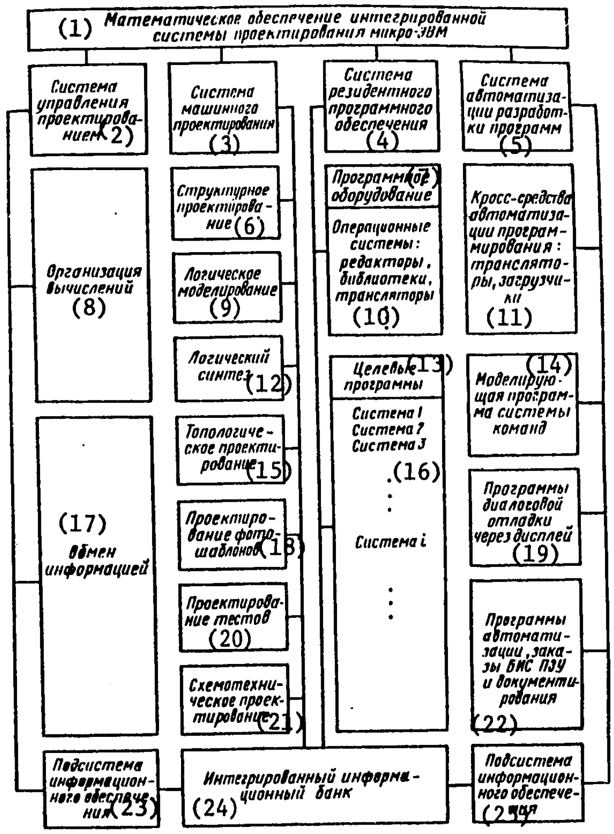


Figure 4.13. Integrated System of Machine Design and Automated Programming for Microcomputers

Key:
(1) Software of Integrated Systems for the Design of Microcomputers;
(2) System for Control of Design;
(3) System for Machine Design;
(4) System of Resident Software;
(5) System for Automation of Program Development;
(6) Structural Design;
(7) Program Equipment;
(8) Organization of Computations;
(9) Logical Modeling;
(10) Operating Systems — Editors, Libraries, Translators;
(11) Cross-Means for Automation of Programming — Translators, Loaders;
(12) Logical Synthesis;
(13) Specific Programs;
(14) Modeling Program of the Command System;

[Key continued next page]

[Key to Figure 4.13 continued]

(15) Topological Design;
(16) System 1, System 2, System, ... System i;
(17) Information Exchange;
(18) Design of Photo Masters;
(19) Programs for Interactive Debugging Through Display;
(20) Design of Tests;
(21) Circuit Engineering Design;
(22) Programs for Automation, Orders for ROM LSIC's and Documentation;
(23) Information Software Subsystem;
(24) Integrated Information Bank.

The program basis of this integration was the information software subsystem of the data archives. It was developed to solve the problem of comprehensive design of LSIC's and is used successfully in both the systems for machine design and in cross-facilities to automate the development of programs for microcomputers delivered on BESM-6 computers. A distinctive feature of the information software subsystem is its successful organization of a two-level data archive which offers great opportunities for working with arrays on magnetic tape and drums using a set of programs [25].

The formulation of a large number of information-control systems based on microcomputers with ROM memory to realize target programs forces the developers of microcomputers to solve the problem of organizing the process of ordering the necessary ROM LSIC's for these systems. It should be observed that the order may involve three types of memory. The first is program equipment devised by the developer of the microcomputer (dispatching system selected for the particular application, the required assortment of standard subroutines, facilities for interaction with the operator, and the like). The second type is standard programs developed by the microcomputer user for the given class of apparatus (for example, various programs for statistical processing of the results of measurements for monitoring-measuring apparatus based on microcomputers). In different configurations these programs may be included in the particular apparatus created on the basis of the microcomputer. The third type is the target programs of the particular type of apparatus.

Because series production of microcomputers with ROM memory imposes the requirement of maximum standardization of the one nonstandard element for this type of production, the ROM LSIC, a number of techniques have been developed in practice that make it possible to reduce the assortment of these nonstandard elements. One of the main techniques is to arrange the corresponding programs in volumes that are multiples of the physical design pattern of the ROM LSIC's of the Elektronika S5 microcomputer. Moreover, a number of programs, regardless of their spheres of application, have been realized in transferrable form, which makes it possible to allocate these ROM memories in any place that is convenient for the user. The same technique is recommended to users to perform standard programs of their own class of applications. This purpose is also served by devising versions of operating systems and other program equipment.

These measures were the program basis for reducing the assortment of ROM LSIC's ordered. The organizational basis for series production was the system of design description of these LSIC's, correlating them with the number of the system in which they are used, the number of the board in the microcomputer, and the number of the place on the board. The number of the ROM microcircuit is determined by the microcomputer user taking into account all three parameters. This is done during the distribution of ROM memory base while developing the system and is transmitted to production in the form of a composition table for the order of microcomputers with the particular ROM memory. This system also permits simple borrowing of programs (and ROM LSIC's) from one system for a microcomputer being used in another system.

An automated system for ordering and producing documents in conformity with the requirements of the Unified System of Design Documentation has been devised and is operated on BESM-6 computers to spare design subdivisions from the cycle of ordering and routine documentation of the information part of ROM LSIC's. This system enables the customer to do this work independently and monitor whether the documentary information corresponds to the required text [26]. An important feature of this system is that it is figured for a user who does not need knowledge in the area of designing photo masters and developing appropriate documentation. The actions of the user amount to sequential performance of operations determined by appropriate instructions. The basic stages of work in the system are shown in Table 4.3 below.

Table 4.3

Principal Stages of Work	Remarks
Inclusion in the list of users served. Writing the ROM composition table into the archive of operational reference information.	
Subdividing the memory field of the ROM memory into coding tables of the matrices of replaceable (information) layers of the LSIC and writing the corresponding information on target programs into the working archive.	Copying information into the working ar- archive and printing it out on demand are permitted.
Recopying information from the working archive to the intermediate archive with printing out the coding tables of the replaceable layers of the ROM LSIC and notice of the fact of recopying.	Printing out the contents of the intermediate archive on demand is permitted. Recopying may be done only with the use of a special password.
Topological design of the replaceable layer of the ROM LSIC and writing information in the archive of control information for drawing a photo master; outputting the set of design documents and information.	

Considering the above and also the practices employed in development and series manufacture of microcomputers for information-control systems, it may be concluded that preparing information for the fabrication of ROM LSIC's is a complex, independent process that begins after specific programs are debugged. This system is based on integration of systems for machine design of LSIC's and automation of development of microcomputer programs, and to carry it out requires an organized and program-based system to insure a contemporary level of series production of microcomputers with target programs in ROM LSIC's.

4.6. Designing Specific Software

The initial stage in the development of the specific software of microcomputer-based equipment is working out technical specifications and composing a functional description (the stage of analysis and design, see Figure 4.14 below). It does not differ from the analogous stage of designing software for minicomputers [27]. The following stages have specific features resulting from the use of control microcomputers to realize the functions of control of data processing and transmission in apparatus, instruments, and systems.

Experience with developing DCPT systems showed that in addition to the general patterns in development of system software on the basis of control computers which manifest themselves during the design of such systems, there are two special features related mainly to their high level of series production and the relatively small volume of software. In the first place, there is an independent stage of design which occurs at an intersection of software and hardware development and defines the boundary between functions realized by software or hardware. In the second place, cross-facilities for automating development during the design of target software have a greater role.

Selecting Hardware-Software Compatibilities. With a given microcomputer speed there are different variations for program realization of functions (minimization of cost, but often with a loss of productivity), use of standard microcomputer functional modules (a certain increase in cost, but also an increase in productivity), or hardware realization (maximum cost and productivity). This problem is solved by an iterative process which it is desirable to converge by the time that debugging of the program target algorithm begins.

The large numbers of microcomputer-based systems and the need to achieve the maximum national economic effect from the systems demand careful work on the questions of optimizing hardware-software compatibilities based on the possibilities of and expenditures for designing the system and the efficiency of their production. This raises, in turn, the question of working out appropriate methodologies and using software based on the principles of selecting optimal solutions from multiple variations to preclude subjective factors in evaluating realization of algorithms for the work of a DCPT system. This question is also timely in the development of such systems based on single-chip microcomputers. It follows from this that it will be important to organize the corresponding work as an independent phase of the process of designing target software.

In this phase of design functional (development of decision-making techniques and performance by the system of its specific designation of data control,

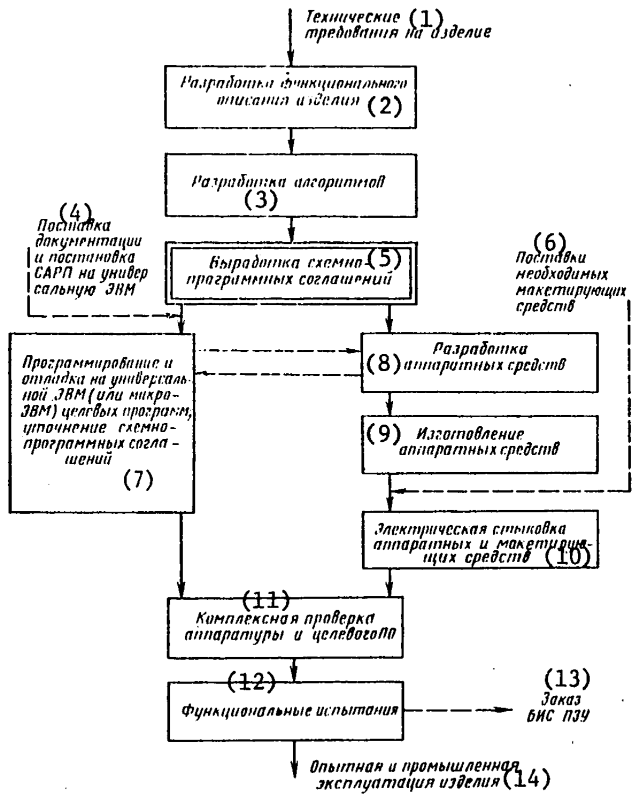


Figure 4.14. Order of Designing Microcomputer Specific Software.

Key:
(1) Technical Specifications for Article;
(2) Development of Functional Description of Article;
(3) Development of Algorithms;
(4) Delivery of Documents and Setting Up Systems for Automated Program Development on General-Purpose Computer;
(5) Working Out Hardware-Software Compatibilities;
(6) Deliver of Essential Modeling Facilities;
(7) Programming and Debugging of Specific Programs on General-Purpose Computer (or Microcomputer), Specification of Hardware-Program Compatibilities;
(8) Development of Hardware;
(9) Manufacture of Hardware;
(10) Electrical Interlinking of Hardware and Modeling Facilities;
(11) Comprehensive Testing of Hardware and Target Software;
(12) Functional Testing;
(13) Order for ROM LSIC's;
(14) Experimental and Industrial Use of Article.

processing, and transmission) and structural (formulating target problems that insure maximum efficiency in use of microprocessor equipment and a high level of technological suitability for design in the system as a whole) problems are solved [28].

In this stage (see Figure 4.15 below) design is characterized by singling out the algorithms of the channels of information exchange between sources (sensors) and microcomputers in the working algorithms of the system.

We will use the phrase channels for exchange of information between the control microcomputer and data sources (sensors) with which the particular microcomputer is linked during formulation of the DCPT system to refer to the software-hardware facilities that carry on digital (coded) data exchange (with essential preliminary processing) between the memory of the control microcomputer and the data source (sensor).

The term "exchange channel" consolidates the concepts of interaction of the control microcomputer with input and output devices for communication with the processor, with facilities for interlinking the human operator and the machine, and with facilities for communication between the particular DCPT system and remote data sources (sensors). This consolidation is accomplished on the condition that any of the above-enumerated variations of interaction has common parameters: amount of information exchange in a unit of time; frequency of exchanges; preliminary processing algorithms.

In order to estimate the microcomputer resources required with respect to speed, memory, and essential hardware support the exchange channels in the system are represented by distinct but interrelated blocks, each of which is realized by one of the feasible software or hardware techniques. The initial calculation is made on the basis of standard means of interaction with the environment available to the user within the given configuration of the microcomputer and assuming maximum use of the program level of realization of the exchange channel. This initial condition is based on the propositions that, for one, the use of these facilities accelerates the development of the article and simplifies further production and, for two, programming is a more highly automated process of design than hardware development.

If total expenditures for all channels in terms of memory volume, speed, or number of standard interaction facilities with the external environment necessary to accomplish the exchange exceed the amounts assigned in advance, the channel that uses the most resources is singled out and the type and configuration of additional hardware necessary for it to significantly reduce these expenditures is determined. The operation can be performed simultaneously for several channels based on the ratio among total expenditures obtained, expenditures for a channel, and assigned expenditures. After this the next estimate of resource expenditures is made and compared with the assigned constraint.

After the hardware-software compatibilities are worked out, the hardware and software parts of the article are developed.

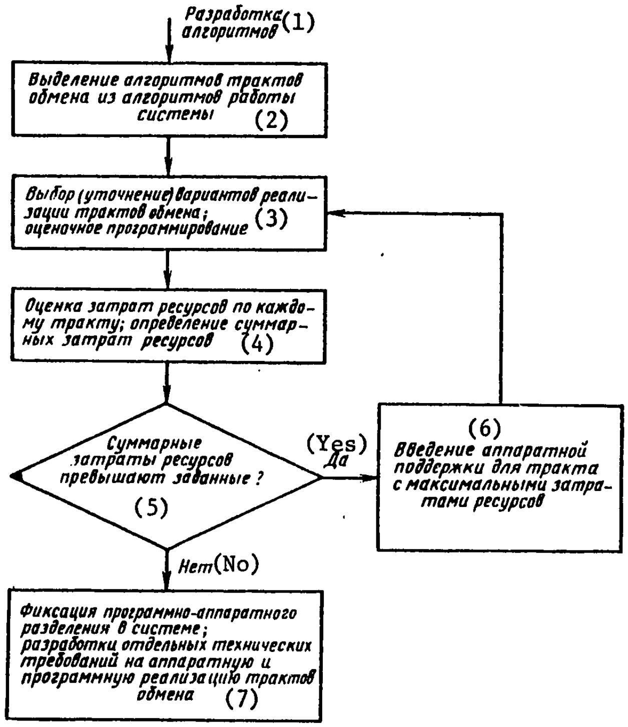


Figure 4.15. Process of Working Out Hardware-Software Compatibilities in DCPT Systems

Key:
(1) Development of Algorithms;
(2) Singling Out Algorithms of Exchange Channels from Algorithms of System Work;
(3) Selection (Specification) of Variations of Realizing Exchange Channels, Evaluation Programming;
(4) Estimates of Expenditure of Resources for Each Channel, Determination of Total Resource Expenditures;
(5) Do Total Resource Expenditures Exceed Assigned Expenditures?;
(6) Introduction of Hardware Support for Channel with Maximum Resource Expenditures;
(7) Establishing the Software-Hardware Breakdown in the System, Working Out Particular Technical Specifications for Hardware and Software Realization of Exchange Channel.

The questions of designing the hardware part of the article are the basic specialization of those development workers to whom this book is addressed, and therefore are not considered here.

Programming and Debugging. Figure 4.16 (below) describes the jobs involved in this stage.

After development of the algorithms of system functioning and preliminary coordination of the hardware-software breakdown of functions the specific software is programmed in machine language, assembler language, or the high-level language with which the microcomputer is equipped. It can be stated with confidence that it is wise for the developers of the specific microcomputer software when formulating problems with a volume of more than 1,000 words to program in autocode, not machine code. The convenience of modifying programs during debugging with these volumes of work will repay expenditures for additional (compared to study of the system of commands) incorporation of facilities for automation programming.

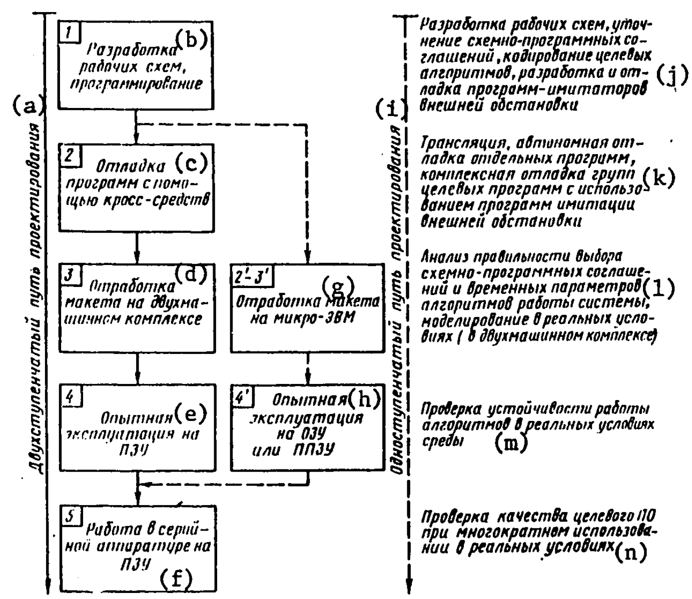


Figure 4.16. Work to Program and Debug Target Algorithms of Systems for Control, Processing, and Transmission of Data.

Key:
- (a) Two-Step Design Procedure;
- (b) Development of Working Circuits, Programming;
- (c) Debugging Program Using Cross-Facilities;
- (d) Development of Model on Two-Machine Complex;
- (e) Experimental Operation on ROM Memory;
- (f) Work in Series-Produced Apparatus on ROM Memory;
- (g) Development of Model on Microcomputer;

[Key continued next page]

[Key to Figure 4.16 continued]

(h) Experimental Operation on Main Memory or Semipermanent Memory;
(i) One-Step Design Procedure;
(j) Development of Working Circuits, Specification of Hardware-Software Compatibilities, Coding Specific Algorithms, Development and Debugging of Programs that Simulate the External Situation;
(k) Translation, Autonomous Debugging of Specific Programs, Comprehensive Debugging of Groups of Specific Programs Using Programs To Simulate the External Situation;
(l) Analysis of the Selection of Hardware-Software Compatibilities and Time Parameters of the Algorithms of System Work, Modeling Under Real Conditions (in a Two-Machine Complex);
(m) Test of the Work Stability of Algorithms in Real Environmental Conditions;
(n) Check of the Quality of Specific Software with Multiple Use Under Real Conditions.

The next stage of design is debugging specific software with cross-facilities for automating program development (see Figure 4.17 below). These design facilities make it possible:

1. To develop programs parallel with devising the hardware framework of the system and its coupling with the microcomputer in a model of the object or a prototype of it.

2. To use the significant (compared to microcomputer) resources of general-purpose or large control computers, above all the memory and external units, for automatic running of a large number of debugging variations, varied documentation of results, accumulation and storage of libraries of programs, and other jobs for which these computers are well-equipped in terms of hardware and software capabilities.

3. To model the external situation using programs realized on the machine being used and connected to the specific programs of the microcomputer through the modeling program of the system of commands; this approach can be used not only in the stage of designing the concrete hardware on the basis of a microcomputer, but also in the process of research or predesign work to choose the work algorithm of the hardware, optimize proposed hardware-software compatibilities, and determine adequate quantitative characteristics of the specific software. This is especially important during the development of hardware with new technical characteristics or based on new working principles (the possibility of an investigation using such facilities is substantially higher than the possibility of reproducing the same results on the debugging stand).

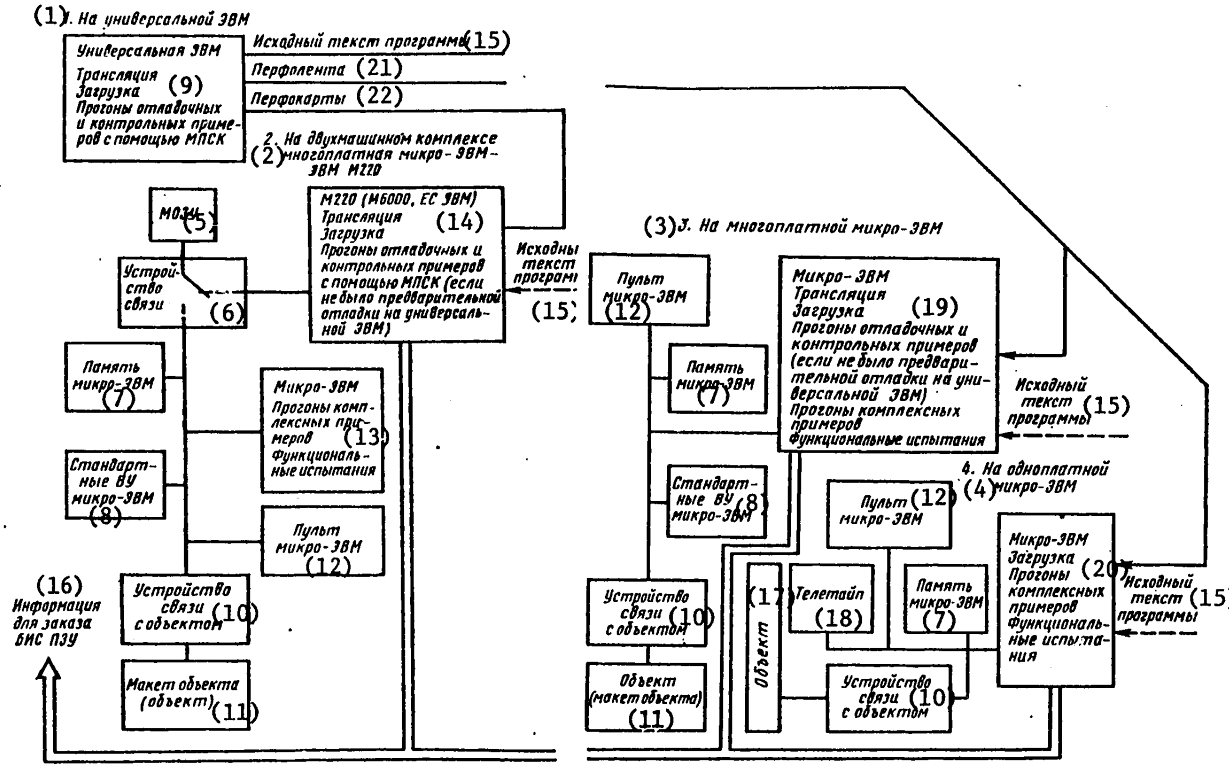


Figure 4.17. Multilevel System for Debugging Specific Software of Equipment Based on a Family of Widely Used Microcomputers

Key:
- (1) On a General-Purpose Computer;
- (2) On a Two-Machine Complex (Multiboard Microcomputer and M220 Computer);
- (3) On a Multiboard Microcomputer;
- (4) On a Single-Board Microcomputer;
- (5) Magnetic Core Storage;
- (6) Communications Device;
- (7) Microcomputer Memory;
- (8) Standard Microcomputer Peripheral Units;
- (9) General-Purpose Computer — Translation, Loading, Running, Debugging and Monitoring Examples Using the Modeling Program of the Command System;
- (10) Device for Communication with the Object;
- (11) Model of the Object (Object);
- (12) Microcomputer Console;
- (13) Microcomputer — Running Comprehensive Examples, Functional Testing;

[Key continued next page]

[Key to Figure 4.17 continued]

(14) M220 (M6000, YeS Computer) — Translation, Loading, Running, Debugging and Monitoring Examples Using a Modeling Program of the System of Commands (If Preliminary Debugging on the General-Purpose Computer Was Not Done);
(15) Initial Text of the Program;
(16) Information for Ordering ROM LSIC's;
(17) Object;
(18) Teletype;
(19) Microcomputer — Translation, Loading, Running Debugging and Monitoring Examples (If Preliminary Debugging on a General-Purpose Computer Was Not Done, Running Comprehensive Examples, Functional Testing);
(20) Microcomputer — Loading, Running Comprehensive Examples, Functional Testing.

For microcomputer-based systems with specific software and a volume of 1-10K words, precluding this stage and employing one-stage designing leads to a great increase in the time required to work out the model on the microcomputer (2'-3') and to a possible loss in time with two-stage designing. This occurs chiefly for two reasons.

In the first place, when developers use one-stage designing of specific software in the period of working out the model on the microcomputer, they have to work with "raw" programs for an extended time. Because interaction with the hardware occupies a significant volume of software in control systems of this class, it becomes difficult to assess the causes of incorrect work: mistakes in algorithms or programs, mistakes in (or failure to observe) the hardware-software compatibilities adopted, and incompletely developed hardware. In addition to the ordinary technical difficulties associated with the search for causes of incorrect system work, natural psychological problems of interaction among specialists in programming and electronics in all stages of design, which determine how quickly a control system based on computers is set up, become more complex. The availability of check variants of specific software developed by means of cross-facilities and simulation programs accelerates the search for the area of the errors by simplifying interaction between programmers and the hardware developers. Hardware test programs developed on the basis of algorithms coordinated with the hardware developers can also serve this purpose.

In the second place, the time for development of the model, which is restricted to concrete periods, and is partially spent solving the problems mentioned above during one-stage design, narrows the range of testing, which increases the number of unidentified errors and the likelihood that the microcomputer ROM memory in series-produced hardware will have to be adjusted.

But when the volume of specific software is within the range of 1K words, these solutions are possible where stage 2'-3' is conducted carefully and there is experimental operation of the apparatus or device on main memory or semipermanent ROM memory (4') for an extended time in sufficient volume.

Unfortunately, it is not possible to give a precise time evaluation of such work because of the large variety of types of equipment, the conditions of conducting stages 2'-3' and 4', the qualifications of developers, and other organizational and technical matters that affect the time and quality of development work. At the same time, experience with these developments on a number of microcomputer-based devices and equipment shows that under fairly favorable conditions the developers of devices and equipment who spend 10-14 months on the first stage of design of microcomputer-based systems can complete the work if the volume of specific software is not greater than 1K words. Furthermore, it is possible to use either two-stage or one-stage designing of the specific software.

Examples of such projects that were conducted in periods of time close to those indicated are the development of the spectrophotometer based on a single-board microcomputer with two-stage design of specific programs and the development of a general-purpose digital regulator with one-stage design of specific programs. These and numerous other developments, including the development of a software user station, demonstrated that the labor-intensiveness of designing 1K words of a multipurpose microcomputer using the indicated facilities (stages 1-4 [4']) takes 10-17 person-months. This figure is reduced by one-fifth to one-half when designing the following types of microcomputer-based apparatus; this is the result of experience with the use of its hardware and software facilities, the debugging system, and institution of a clearcut procedure in the design process.

Comprehensive debugging. The decisive stage in the preparation of specific software from the standpoint of high-quality work by series-produced microcomputer-based equipment is working out programs on the microcomputer along with hardware framing within the model or experimental equipment (see Figure 4.17 above). This stage provides the final test of whether the ideology of constructing specific programs and the selection of hardware-software compatibilities were correct and also refines the principal time characteristics of microcomputer work within the equipment. When describing this stage it is advisable to compare hardware and software facilities for doing it.

These facilities include: a microcomputer equipped with resident facilities for automating program development (a translator from autcode, a loader, and software-hardware support for monitoring program codes); and, a two-machine "microcomputer — M220" (or M6000 or YeS computer) complex with cross-facilities for automating program development (translator from autocode, loader, and modeling program of the set of commands).

There are two advantages to using only a microcomputer interlinked with a model of the object or an experimental prototype: the absence of expenditures for setting up a two-machine complex and putting the appropriate software on it; the speed with which work begins on the model, which is determined entirely by the time required to deliver the appropriate microcomputer and connect it to the framing hardware. For enterprises whose technical policy is moving toward development of microcomputer-based apparatus, devices, and control systems, however, this way is acceptable only for initial development using microcomputers and only within the restrictions mentioned above.

The only way to achieve the maximum degree of reliability in debugging specific microcomputer software in a fairly limited time is to use a two-machine complex that combines elaborate hardware typical of general-purpose computers (fast input-output units and external memory) with the functioning of real microcomputer processors and input-output units together with a model of the object. The cross-facilities put on the "old" computer of a two-machine complex provide fast and optimal modification of the programs of specific software during the process of their testing and correction within a two-machine complex. These facilities can also be used for cross-debugging of programs during the design of specific software. But as practice has demonstrated, enterprises that are developing several types of microcomputer-based articles begin to use such a complex almost continuously for conducting particular stages of hardware debugging: interlinking the microcomputer of the complex with the model of the object or the experimental unit; debugging modeling programs in the real-time scale of the external situation; testing the work of specific software and functional testing of the model (experimental unit) to make the decision on experimental operation of the articles using ROM memory or main memory (or semipermanent ROM memory).

Therefore, in those cases where it is possible to use cross-facilities for automating the development of specific software on a general-purpose computer that is not part of the two-machine complex, it is always advisable to do so to reduce the time of stages 1, 2, and 3 of development of the microcomputer-based articles.

In completing our description of this stage, we must note that its final phase is functional testing. This term is used to describe a new process in the design of apparatus, devices, and control systems. Its appearance resulted from the fact that in the construction of microcomputer-based articles most of the functions or the main ones are accomplished by program techniques, while the remainder can be done only by hardware framing whose proportion will steadily diminish as the speed of multichip microcomputers increases and the practice of designing single-chip microcomputers is introduced. Therefore, testing programs that realize the working algorithms of the article before they are reproduced on the ROM memory, the regular medium, is becoming a distinct and essential part of the process of designing specific software, and the nature of execution of the hardware framing at this moment no longer plays a significant part (the only important thing is that it have complete functional correspondence to the planned future realization). Moreover, the testing of specific software, and of the article or a model of it in practice, should be done by those programs and methodologies which must be used for stand testing of experimental units of the article (on ROM memory, main memory, or semipermanent ROM memory with specific software).

In view of its completeness from the functional standpoint and a certain conditional quality from the standpoint of the framing and using auxiliary debugging facilities (of the two-machine complex), such testing is called functional testing (as distinguished from testing experimental units, where the form of hardware and software is determined by existing standards).

Following the results of functional testing conducted on the model after its development (3 or 4, 2'-3') a decision is made on transmitting information on the specific software for the manufacture of microcomputers with ROM memory for experimental operation and/or stand testing of the microcomputer-based articles or preparation of an article for conducting these stages with storage of information in the main memory or semipermanent ROM memory of the microcomputer. Stage 3 usually concludes with transmission of information for the fabrication of ROM LSIC's because the facilities for conducting this stage insure high reliability in the specific software. At the same time, the results of functional testing conducted upon completion of the work and stage 2'-3' do not always testify to the possibility of fabricating ROM LSIC's, and require experimental operation on such an information medium as the main memory or semipermanent ROM memory for a more detailed check of the article in different work regimes.

The care taken in preparing specific software using all debugging facilities becomes an even more important factor when single-chip general-purpose microcomputers built into the equipment are used. The correction of specific software in multichip microcomputers requires redesigning of replaceable layers and the replacement of all or several ROM LSIC's,which are just one of the functional elements of the microcomputer. When a single-chip microcomputer is used the analogous correction process is a more complex "surgical" operation with all the conditions of fragmentation of design of particular elements of this chip.

In conclusion, several deductions can be made.

1. The use of cross-facilities for automating programming is not only a technique to improve the reliability of programs, but also a way to reduce the design time for microcomputer-based articles by combining stages of the development of specific software with other stages of article design.

2. The debugging of an article using a two-machine "microcomputer — general purpose computer" complex (with a large control computer) insures high-quality specific software in series-produced equipment, beginning with the first models.

3. The importance of using all facilities for debugging specific software increases with the use of single-chip computers in apparatus, instruments, and systems [29].

Chapter 5. Key Trends in the Application of Elektronika S5 Series Microcomputers

5.1. Principles of the Use of Microcomputers

At the present time, two gradually diverging trends are observed in microcomputer engineering.

The first trend, which is represented by the RDR-11/03, Elektronika-60, and Elektronika-NTs computers has been called the "microminicomputer" trend. These machines are characterized by an endeavor to make maximum use of the opportunities presented by a high level of LSIC integration to increase computing capacities, microcomputer memory volume, and the speed of the input-output channels.

The second trend typically uses a growing degree of LSIC integration to create functionally complete computer configurations with a minimum number of functional modules or LSIC's. This trend developed with the appearance of single-board microcomputers and functional modules for them. Single-chip microcomputers are a logical extension of this trend.

It is very important to choose the correct family of microcomputers when beginning to design an instrument, aggregate, or system based on microcomputers. In this connection all applications of microcomputers can be classified in the following four groups:

1. The microcomputer is used in a particular system in place of a minicomputer employed (or planned for use) earlier. The chief advantage of the microcomputer in this case is its very low cost and high degree of series production, which makes it possible to use a control computer in places where this was practically impossible before owing to economic considerations or the impossibility of obtaining the required number of machines. Thus, it can be said that the microcomputer is a means to setting up inexpensive and large-scale automated systems for industrial processes and other systems of similar functional design. It is important here that the microcomputers have sufficiently elaborate peripheral units. This group of applications is characteristic of the Elektronika-60, Elektronika-NTs, and Elektronika-S5-02(01) microcomputers. It should be observed that the relatively low speed of the latter does not limit its application in many cases because long years of experience with setting up automated control systems for industrial processes have demonstrated that the speed of the minicomputers included in them, machines such as the Elektronika-100 and Elektronika-K200 is far from fully utilized.

2. The microcomputer is used as a computer built into a particular device or instrument, and the computing capability of the machine is an important characteristic while no additional functions related to controlling equipment work are assigned to the machine. The Elektronika S5-12(11) microcomputer can be used as a computing unit if its speed and main memory volume are adequate in the specific case. But this application is not typical for this microcomputer; it is more characteristic for the LSI-11 single-board microcomputer of the DEC company [30], which has fairly high speed and comparatively large main memory volume, but requires the use of additional equipment to organize interaction with the controlled equipment.

3. The microcomputer is used as a supervisor, a built-in block to control a machine tool, aggregate, or instrument. Such applications usually produce the greatest advantage from the standpoint of reducing the labor-intensiveness of fabrication and cost of the equipment, improving its reliability, and giving it new qualities in fast processing of measurement results, convenience of operator work, and the like. The most important quality of the microcomputer for this group of applications is maximum functional completeness in a minimum configuration. In this respect, the single-board and single-chip models of the Elektronika S5 family of microcomputers are superior to other microcomputers because, as has already been observed, they have on one board (one chip) not only a processor and main memory, but also parallel interface circuits to 32 inputs and 32 outputs, interrupt circuits, timers, and main memory. It would take at least four boards to realize such a configuration from, for example, LSI-11 modules.

4. The microcomputers are used within a hierarchical system of computing facilities, in decentralized systems. We know that decentralization of this type is a way to increase the flexibility and vitality of a system, while the low cost of microcomputers has eliminated the main obstacle to setting up such systems. It is necessary to decide which level of the hierarchy is determining when selecting the family of microcomputers to construct such a hierarchy. Thus, if it is necessary for the hierarchy to have minicomputers with large computing capabilities and an elaborate system of magnetic tape and disk external memory, but the lowest element of the hierarchy is not too critical with respect to volume and cost of the microcomputer included in it, preference should be given to machines such as the Elektronika-60, which is compatible with the SM-3, SM-4, and Elektronika 100/16I minicomputers.

By contrast, if the principal objective of the hierarchical system is to achieve maximum economy in data control and processing at the lower level and the requirements for speed and volume of processing information at the second level of the hierarchy are not too great, preference should be given to the series of signed for setting up small configurations. The Elektronika-S5 family of microcomputers makes it possible to construct such systems. For example, the Elektronika S5 single-board or single-chip microcomputers can operate as supervisors in equipment, and the Elektronika S5-02 machine can control a group of equipment and, if necessary, be linked by telephone and telegraph channels with a general-purpose computer located at a significant distance away.

5.2. Survey of Microcomputer Applications

In beginning our review of the basic applications of Elektronika-S5 microcomputers in different sectors of the national economy we propose the following procedure to the reader. We will try to identify the distinctive functions given to microcomputers which are fairly general in character and, with some modification, occur in many applications. Let us review some "characteristic functions" of microcomputers.

1. Replacement of rigid logic with program logic. This is one of the most obvious ways to use microcomputers. It basically involves building equipment based on microcomputers that generally coincides in functions with series-produced equipment with medium and low levels of circuit integration, discrete electronic components, relays, or mechanical elements.

2. Building control elements for automatic regulation systems. The application of digital control machines in servo-systems has long had a solid theoretical basis. Thanks to their low cost and high degree of series production, microcomputers provide a technical basis for constructing this type of servo-mechanism.

3. Constructing digital automation circuits. As a general-purpose programmable element with main memory and the capability of working in real time, the microcomputer makes it possible to realize almost any logical expression where parameters coming from outside and ongoing time may operate as variables.

The possibility of realizing functions (2) and (3) simultaneously by means of microcomputers is also extremely important.

4. Providing interaction with the operator. The inclusion of a microcomputer in a particular piece of equipment, measuring device, machine tool, or data transmission unit makes it possible to give this equipment means of communication that only large computer-based systems formerly had: representation of information in the most convenient and visible form; monitoring the correctness of operator actions; suggesting the order of actions, and so on. This provides such a substantial increase in the productivity and error-free quality of operator work that in many cases it will be the primary, and even sole, function of the microcomputer that is built into the equipment.

5. Interface organization. This function is assigned to a microcomputer built into equipment that is capable of working with a particular collective-use line. In this case the system of conductors, the set of service signals, and the time chart of data exchange for such a collective-use line differ significantly for different fields of engineering. Examples of such standard collective-use lines are the Camac standard [31], the JEE-488-75 instrument interface, and various communications interfaces.

6. Conversion of information during its transmission (reception) through communications channels. This microcomputer function may amount specifically to introducing (removing) noise-proof coding during the transmission (or receipt) of messages, carrying out a procedure to compress information to increase the carrying capacity of the communications channel, organize transmission of variable parts of formalized messages, and the like.

7. Decentralization of computations and control. Without touching on the general problem of organizing multimachine complexes based on microcomputers, we will simply observe that the small size and low cost of microcomputers make it possible to bring the machine very close to the source of information and control object, and ultimately to build microcomputers into them. Together with functions (5) and (6) this is one of the ways to save expenditures on communications lines, and improve the flexibility, rearrangeability, and vitality of DCPT systems.

The functions reviewed above do not comprise all functions that are presently given to microcomputers, to say nothing of those that will be in the future. The very same functions may be realized in different equipment by software or by functional modules that supplement the capabilities of a single-board microcomputer. Some of these functions may be realized on circuits with a medium degree of integration because microcomputers do not have adequate speed. Finally, when realizing certain functions they may all be done by one microcomputer or a particular machine may be assigned for each function, even if there is no problem with speed, composition of memory, and input-output devices. The choice of the particular engineering solution is a very important part of the process of designing microcomputer-based equipment.

Let us consider how these problems were solved in building a number of instruments and systems based on Elektronika S5 microcomputers.

5.3. Data Transmission Equipment

This class of equipment is characterized by extensive use of third-generation electronic components. Therefore, the basic problems facing this equipment are reducing the labor-intensiveness of manufacture, raising reliability, improving capability for series production, and reducing size and power consumption; these can be solved well by using the microcomputer as the principal component.

The programmed user station based on the Elektronika S5-01(02) microcomputer is a characteristic example of building such equipment [14, 32] (see Figure 5.1 below).

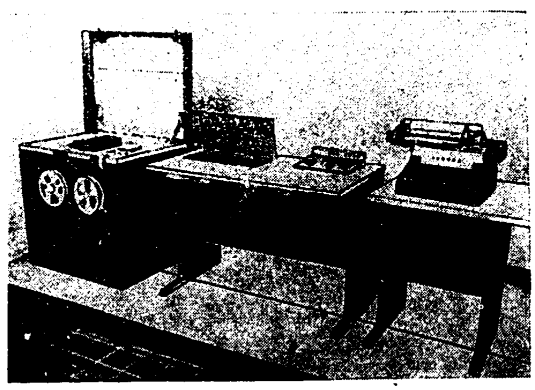


Figure 5.1. Programmed User Station

The purposes of setting up such a user station were: a significant reduction in the labor-intensity of manufacturing user stations; realization at the programmed user station of a structure adaptable to changes in algorithmic requirements for a particular article; application of concepts which would make it possible to utilize continued growth in the technological parameters of microcomputers (level of integration and speed) without revising the article as a whole.

The user station constructed on the basis of microcomputers is designed to transmit data by telephone and telegraph lines (separate and switchable) at speeds of 50, 100, and 200 bits per second. The microcomputer in the user station provides information exchange with Ridm R40 and Perfom R35 punched tape input-output units, a Konsul-260 electric typewriter (GOST [State All-Union Standard] 13052-74) to print out messages and prepare and edit data; a unit to convert signals at the C2 interface (Modem-200 TG [expansion unknown] signal convertor), and a user station control console.

The reliability of transmission by characters is no worse than $2 \cdot 10^{-6}$ in a communications channel with a probability of error of 10^{-3} per character. The block data transmission technique (with 15, 30, or 45 characters per block) is used with waiting for a response to the transmitted block. Data is transmitted with a matrix even parity check, and errors are corrected by repeating transmission of the block. The basic requirements of the national data transmission network are met in the programs for interaction with a communications channel in the microcomputer.

Figure 5.2 below presents a structural diagram of the programmed user station. Its working algorithms are realized by means of programs and microprograms with a total volume of 21K words and modules for communication with the console of the user station (digital input-output boards), the Modem (board for coupling with the signal convertor), with the electric typewriter (board for control of typewriter mechanisms), and with the tape punch (board for coupling with the photo reader and punch).

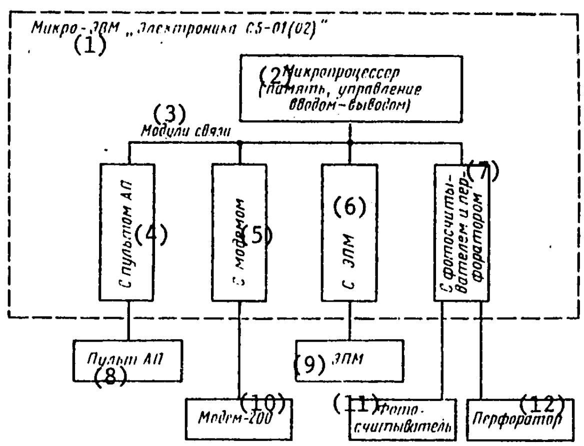


Figure 5.2. Structural Diagram of the Programmed User Station

Key:
(1) Elektronika S5-01(02) Microcomputer;
(2) Microprocessor (Memory, Input-Output Control)
(3) Coupling Modules;
(4) With User Station Console;
(5) With Modem;
(6) With Electric Typewriter;
(7) With Photoreader and Tape Punch;
(8) User Station Console;
(9) Electric Typewriter;
(10) Modem 200;
(11) Photo Reader;
(12) Tape Punch.

There are several distinctive features to the realization of these algorithms. In the first place, flexible distribution of user station functions among different levels of realization (program, microprogram, and hardware) made it possible to find an efficient solution to the problem of minimizing expenditures for performance of these functions. Moreover, during review of the work algorithms of the user station groups of problems were singled out whose modification within the limits of the group could be independent of other groups. The most typical examples of such groups of problems are: control of problem-solving and exchange with external units done in the form of a dispatcher system; work on a communications channel defined by a type of protocol adopted within the particular data transmission network, and preparing and editing messages before transmitting them on the channel. In the second place, the hardware used in the user station is standard models for coupling with peripheral units included in the Elektronika S5 family. In the third place, during development of user station software performance of the functions of controlling problem-solving and exchange with external units were assigned to a standard dispatching system belonging to the microcomputer and oriented to such an assortment of peripheral units.

The construction of a programmed user station based on the Elektronika S5-01(02) microcomputer confirmed all the advantages of developing this type of DCPT system on the basis of multipurpose microcomputers compared not only to "rigid logic" but also (with respect to a number of key parameters) compared to microprocessor sets or specialized microprocessor systems.

Among these advantages we must list first the fact that in this realization the user station is a standard microcomputer program to work as the terminal equipment of a data transmission network. This means that the development of the user station amounts to formulating the specific software using everything necessary from the set of general microcomputer software accumulated at the particular moment (recoding and editing programs, dispatcher systems, dialog programs, and so on); then it involved design formulation of the elements of the microcomputer together with the required peripheral units. This is the principal advantage of using such basic elements as compared to any others. In addition, this approach to development provides a rapid decrease in the cost of the article as a whole through reduction in the cost of its basic element, the multipurpose microcomputer whose capability for series production is substantially higher than that of the particular article. This makes it possible to continuously improve the technical-economic characteristics (cost and reliability) of the particular generation of microcomputers.

These methods of realizing algorithms also provided an opportunity to use certain specific user station programs (exchange on a communications channel in networks with analogous protocol, data preparation, monitoring data transmission, and the like) in other DCPT systems. And making them in a form independent of the initial address (in movable form) and correlating the appropriate program volumes to the design unit (ROM LSIC) makes it possible to borrow existing ROM LSIC's directly.

Finally, the technical-economic parameters of the user station can be improved by the use of microcomputers of a particular family developed on the basis of new basic elements. This improvement can be accomplished in two ways: correlating existing specific and standard software with the new microcomputer or devising a fundamentally new article with characteristics that improve all the parameters of the user station built with the preceding generation of microcomputers.

The first method produces a reduction in the number of microcomputer elements used in the user station (the number of LSIC's and boards) while also reducing the cost of each element and expanding such technical characteristics as the increase in transmission speed on the communications channel and the introduction of additional procedures. The volume of reprogramming in this case does not exceed 10-15 percent of the total volume of the programs.

The second method assures maximum use of new microcomputer characteristics that make it possible to significantly decentralize functions by employing the multimachine method of realizing user station algorithms (the latter is particularly important where single-chip microcomputers are used in the systems).

Therefore, use of the Elektronika S5-21 microcomputer in the above-described user station permits us to consider that it can be used in two ways.

1. It is possible (see Figure 5.3 below) to take the Elektronika S5-21 microcomputer and transfer all available specific software to it (user station algorithms realized on the microprogram level of the Elektronika S5-01 microcomputer should be done at the program level). Communication with peripheral

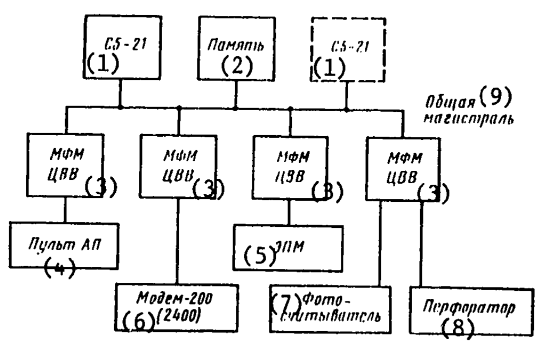


Figure 5.3. Centralized Structural Diagram of a Programmed User Station Based on the Elektronika S5-21 Microcomputer.

Key:
(1) Elektronika S5-21;
(2) Memory;
(3) Digital Input-Output MFM;
(4) User Station Console;
(5) Electric Typewriter;
(6) Modem-200 (2400);
(7) Photo Reader;
(8) Tape Punch;
(9) Common Line.

units is organized through digital input-output modules. If the speed is inadequate a second microcomputer is connected in and given only the functions of exchange with peripheral units.

2. It is plausible to devise a completely distributed system that realizes user station algorithms with a set of MFM's [microprocessor functional modules] devised to expand the functions of the Elektronika S5-21 microcomputer and also several Elektronika S5-21 microcomputers that perform the functions of specialized MFM's and machines to process essential (specific) information (see Figure 5.4 below).

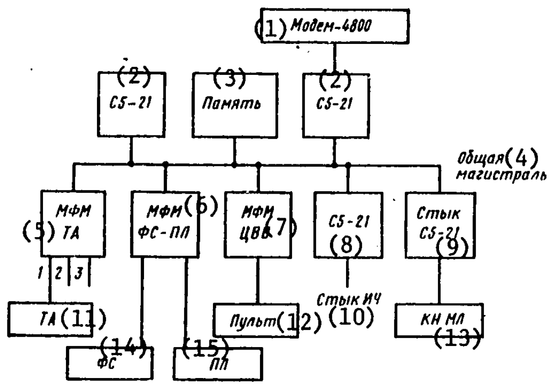


Figure 5.4. Distributed Structure of a Programmed User Station Based on the Elektronika S5 Microcomputer.

Key:
(1) Modem-4800;
(2) Elektronika S5-21;
(3) Memory;
(4) Common Line;
(5) Teletype MFM;
(6) Photo Reader-Tape Punch MFM;
(7) Digital Input-Output MFM;
(8) Elektronika S5-21;
(9) S5-21 Interface;
(10) ICh Interface;
(11) Teletype;
(12) Console;
(13) Magnetic Tape KN [Expansion Unknown];
(14) Photo Reader;
(15) Tape Punch.

The availability of facilities for automating program development and system debugging that have been tried and mastered by the developers of user stations make it possible during establishment of the technical specifications of the new generation of microcomputers of this family (that is, 2-3 years before their series production) to work out and model the structures and software of modifications of user stations employing these microcomputers, which greatly reduces the time between the start of development of the basic element and production of the new article when compared with the sequential method of designing. This approach was successfully tested during the development of the above-described user station.

5.4. Instrument Making

Microcomputers as measuring instruments are used for the following purposes: to improve the precision of measurements; to automate the work of the instrument and the measurement process; to increase the productivity and reduce the errors in operator labor; to introduce the functions of quick processing of measurement results; to provide the possibility of using the instrument within a measurement system; to improve the economic characteristics of the instrument — reduce labor-intensiveness of manufacture, size, and power consumption.

Let us consider how these objectives are achieved when the Elektronika S5 microcomputer is used in various instruments.

A radio measurement instrument based on a single-board microcomputer. As an example let us consider the application of the Elektronika S5-11 microcomputer in a multichannel generator of code sequences, giving special attention to automation of operator interaction with the instrument and the organization of the instrument interface for work within a measurement-information system — two microcomputer functions that are very characteristic of its application in measurement instruments [33].

Code sequence generators are designed to test information systems, communications channels, and multifunctional digital units by outputting binary code sequences up to several thousand bits in length with a frequency up to several dozen megahertz on several channels simultaneously. Two types of problems arise with the construction of such an instrument. The first relates to insuring the high-frequency characteristics of the output cycles of the instrument and is solved by employing high-speed switching elements. The second arises from the necessity to realize very intricate logic in the work of the instrument and from the abundance of raw data that must be fed to it before work begins. It is this type of problem that determined the following basic objectives in development of such an instrument based on microcomputers: organization of operator interaction with the instrument; interlinking the instrument with an LKP [collective use line]; maximum automation of instrument work; self testing of the instrument and its particular assemblies during the operations process.

The high-frequency part of the instrument (see Figure 5.5 below) consists of a high-frequency block, a buffer memory unit, and output stages. Information on the work regimes of the instrument and the code sequences that have been formed is stored in the long-term memory and does not disappear when the power source of the instrument is shut off. Information is outputted from the main memory of the generator through the buffer memory. All these blocks are interlinked with the microcomputer, but the work of each of them and transmission of data among them is controlled by hardware. The instrument panel and console are interlinked with the microcomputer.

Let us consider the basic functions of the microcomputer in greater detail.

Operator interaction with the instrument includes feeding parameters that determine the work regimes of the instrument from a keyboard; representing

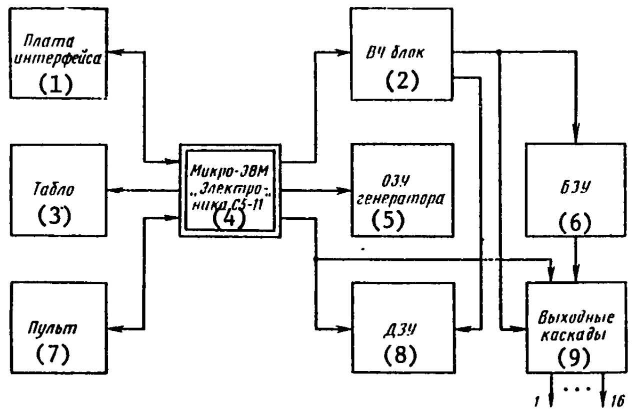


Figure 5.5. Structural Diagram of a Code Sequence Generator.

Key: (1) Interface Board;
(2) High-Frequency Block;
(3) Panel;
(4) Elektronika S5-11 Microcomputer;
(5) Main Memory of Generator;
(6) Buffer Memory;
(7) Console;
(8) Long-Term Memory;
(9) Output Stages.

---

parameters and instrument work regimes being fed on the panel; monitoring parameters and code sequences being fed, indicating the results of monitoring, and stopping the work of the instrument when errors are detected in the information.

Unfortunately, the instrument does not adequately provide for the possibility of suggesting the essential sequence of actions to the operator. This is because of the limited capabilities of the panel that is used.

The use of a compact keyboard, replacing several dozen multiposition disk switches and tumblers became possible only thanks to program realization of interactive functions, and indeed it is not advisable to realize these functions themselves with "rigid" logic (not using microcomputers). It is very important that the digital information is fed to the instrument from a keyboard and represented on the panel in decimal form and the conversions from decimal to binary and binary to decimal form which are central for work of the instrument are done by a program.

Program monitoring of the procedure for feeding parameters and shaping code sequences pursues two purposes.

In the first place, it precludes mistaken actions by the operator in controlling the work of the instrument, selecting work regimes, and feeding parameters. This is very important because these preparatory procedures alone require the operator to press up to 200 keys on the instrument control board. Undetected errors lead to incorrect measurement regimes and, ultimately, to significant unproductive expenditures of time during instrument operation.

In the second place, monitoring the meaning of each parameter fed, comparing it with boundary values, and checking permissible ratios among several parameters that are fed provides a check not only on the correctness of operator actions but also the correctness of the raw data, which determines the regimes of the measurements made by the instrument. It is clear that these objectives cannot be accomplished in principle without a computing machine, a built-in micro-computer.

Realization of interaction functions is quite universal and can be recommended for various measuring instruments. Their development is being furthered by alpha-numeric indicators, for example the Kvant video monitor with a display adaptor functional module (the Kvant was described in Chapter 2). This makes it possible to carry on diaglog in natural language, output for the operator, a diagnosis of malfunctions and information errors with comments, to suggest the order of performance of manipulations on the console, and so on.

The functions of organizing the interface include: electrical coupling with the LKP; realization of the algorithm for instrument interaction with the LKP; insuring the possibility of remote control of instrument work; insuring transmission of arrays of coded sequences on the LKP from the central computer or other data source.

The collective-use line (LKP) with which the instrument is linked corresponds to international standard JEE-488-1975. More than 300 types of measuring instruments capable of working together in measurement systems are produced following this standard [34]. The introduction of this standard is taking place parallel with development of microprocessor engineering, and its realization in various instruments involves the use of hardware-software techniques in different proportions — primarily by hardware when it is necessary to achieve high speeds of data exchange and mainly by software when it is necessary to conserve equipment and there are no rigid speed requirements.

The standard line contains 16 signal lines, eight of which serve for two-way transmission of coded messages represented by sequential bytes whose binary bits travel along the line in parallel. Further, three lines serve to transmit a sequence of interrelated signals to establish communications and control the transmission of a byte of information, while five lines are for general control of the line, including signals for clearing, remote control, service interrogation, and the like.

The standard envisions an asynchronous mode of data transmission and combined work by instruments with different speeds. Up to 15 instruments can be joined together into a system at one time.

In the generator we are considering the interface functions are given almost entirely to the microcomputer, and only the coordination of the levels of signals and power of output circuits is done on a few dozen integrated circuits with low and medium degrees of integration. Such an economical solution proves possible because the low speed of receiving raw data on an LKP, 100 bytes per second, does not affect the operating characteristics of the instrument (its productivity and frequency characteristics).

Feeding information to an instrument through an LKP, alongside the possibility of working in a measurement system, makes it possible to feed code sequences automatically, not manually. This greatly increases the productivity of the instrument and the reliability of the information being fed.

The function of organizing the instrument interface under consideration is also very general for a broad class of instruments. It is wise to include it in most of the new instruments being developed, even if present needs for such interconnection are not obvious. The new quality which the instrument receives in this case may open up entirely new spheres of application for it.

In many cases the realization of the interface will be defined as the primary and even the sole function of the microcomputer in the instrument; the approach is especially timely where single-chip microcomputers are used.

The functions of automating the work of the instrument are in large part synonomous with the preceding functions; the instrument can also realize automatic selection of the frequency range.

The functions of instrument self-monitoring include checking the correct operation of particular assemblies of the instrument (the microcomputer itself and discrete assemblies) by feeding standard test combinations and testing the correctness of the information being generated.

As already indicated, all of these functions have been realized on a single-board microcomputer without additional functional modules. The programs have a volume of 1,024 words. All that was needed for further development of instrument characteristics in this case was to increase the volume of the programs.

The change to an Elektronika S5-12 model with an ROM memory volume of 2,048 words made it possible to use software to control the long-term memory, realize a large part of the internal organization of the instrument, and expand self-monitoring functions. Increasing memory volume by 1,024 words as a whole made it possible to save more than 500 integrated circuit bodies.

An optical instrument based on a single-board microcomputer and a set of functional modules. As an example let us consider the application of the Elektronika S5-12 microcomputer in the SF-29 spectrophotometer (see Figure 5.6 below). We should give special attention to two other functions of built-in microcomputers, functions that are very important for their use in measuring equipment — quick processing of measurement results and maximum possible automation of instrument work [35].

The spectrophotometer is a fairly complex physicomechanical instrument with an elaborate electronic part. It is designed to measure the spectral transmission coefficients of liquid and solid transparent substances in the working domain of the spectrum. These coefficients define physico-chemical parameters and the quantitative composition of the substance. The principal component parts of the instrument are (see Figure 5.7 below): a monochromatic light source (monochromator), light filters, a sample feeding unit, a detector cell, a panel, an

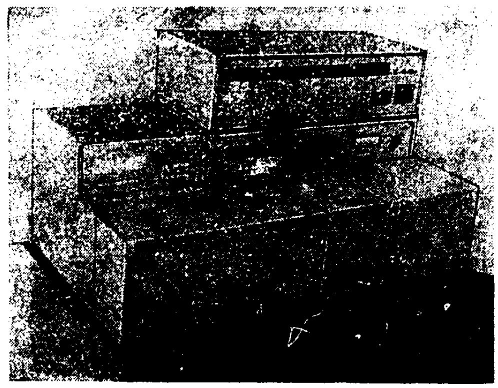


Figure 5.6. The SF-29 Spectrophotometer.

automatic recorder, an Elektronika S5-12 microcomputer, an Elektronika S5-121 alphanumeric printer, a keyboard, an Elektronika S5-123 teletype microprocessor functional module, and a teletype.

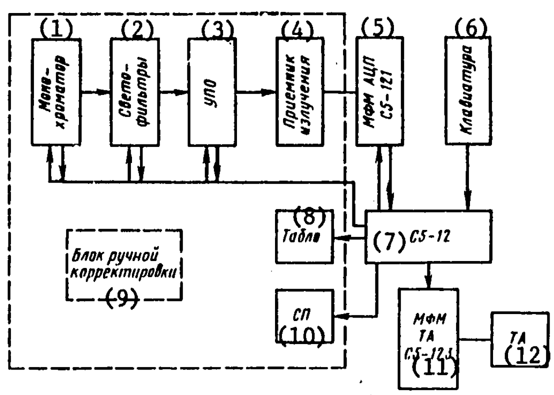


Figure 5.7. Structural Diagram of a Spectrophotometer Based on the Elektronika S5-12 Microcomputer.

Key:
(1) Monochromator;
(2) Light Filters;
(3) Sample Feeding Unit;
(4) Detector;
(5) S5-121 MFM Alphanumeric Printer;
(6) Keyboard;
(7) SF-12;
(8) Panel;
(9) Manual Correction Block;
(10) Automatic Recorder;
(11) S5-123 MFM Teletype;
(12) Teletype.

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

The spectrophotometer works as follows: the monochromator emits a beam of light in conformity with the available wave length. The light passes through the light filters and the sample being tested, which is located in the sample feeding unit, strikes the detector, which gives out a signal whose amplitude is proportional to the transmission coefficient light. The signal goes from the detector to the alphanumeric printer, from where it is received by the microcomputer, processed, and outputted to the panel, automatic recorder, and telegraph. The microcomputer sees that the monochromator uses the required wavelength, sets up the corresponding light filters, controls the programmed sample feeding unit, receive the signal from the alphanumeric printer, processes it in conformity with a program assigned by the operator, and outputs processing results to the panel, telegraph and automatic recorder. Raw information can be fed by the operator through the manual keyboard or telegraph.

The microcomputer in the spectrophotometer is used for these purposes: automation of instrument work; quick processing of measurement results; remote control of instrument work and recording measurement results; organizing operator interaction with the instrument and instrument self-monitoring.

The function of automating the work of such instruments, which involve high-speed, precision processing of movements to measure the working wavelength in the process of measurement,and a fairly complex procedure for moving cassettes with samples and light filters in conformity with an assigned sequence, is especially important.

For this reason, in the first place, the microcomputer controls the servo-system at variable speed depending on the magnitude of the necessary movement. The feedback circuit uses a 17-bit convertor to change the angle of turn to binary codes. In the second place, the microcomputer performs the role of a time-program unit that issues commands to measure samples at assigned time intervals. These briefly described functions of the microcomputer related to automation of instrument work are quite characteristic and in some cases can be primary for work within an instrument assemblage. This is especially important for single-chip microcomputers.

The functions of quick data processing in similar measurement instruments may be developed almost without limitation because ultimately spectrophotometric investigation of the properties of materials may be concluded by processing the data on general-purpose computers. A spectrophotometer that is not linked to a computer does not permit any preliminary conclusions directly during the measurement process and cannot be included, for example, directly in an industrial line in complex contemporary production facilities. For many models of spectrophotometers, therefore, an interlink with a minicomputer is envisioned. This makes it possible to carry out fairly extensive data processing right at the point of measurement. However, such solutions can be used only with the most expensive models of spectrophotometers, because of the comparable cost of the instrument itself and the minicomputer.

The use in the instrument of a built-in microcomputer which replaces a fairly large volume of discrete elements and integrated circuits necessary to realize the functions considered above enables us to speak of the practically "free" introduction of quick processing functions in the instrument.

The SF-29 spectrophotometer realizes the following algorithms for quick data processing: computation of a smoothed quantity; computation of the first and second derivatives from the smoothed quantity; computation of the integral for the spectrum; and fixation of extreme points on the teletype and automatic recorder.

An algorithm for noise-free measurement is also realized. It enables the instrument to operate in noise conditions. Such an algorithm can be realized only on the condition that the microcomputer is a built-in, unremovable part of the instrument.

It follows from all that has been said that an increase in microcomputer resources, above all memory volume, makes it possible to expand the range of algorithms for quick data processing and to realize a specialized language in the instrument to simplify the procedure for operator interaction with the instrument.

The functions of remote control and registration of results are provided by a functional model for control of the teletype and a program for exchange with the teletype. They make it possible not only to output measurement results to the printer but also to feed raw data for automatic control of the measurement program. The teletype is not the only way in which raw data can be fed. By using the teletype channel the instrument can be connected to a computer to which measurement results are transmitted for storage and subsequent processing, and then the program of this work is returned to the instrument. It can be considered that the availability of output to a telegraph channel is one of the ways to realize the instrument's interface functions by means of microcomputer.

The functions of self-monitoring and organizing dialog in the spectrophotometer do not differ in principle from the analogous functions in the code sequence generator.

The described instrument can be switched to new, more sophisticated models of Elektronika S5 microcomputers for basically technical-economic purposes, in addition to the purpose of expanding memory volume for calculation problems. These technical-economic purposes may be reducing the number of functional modules (for example, by virtue of the fact that the digital input-output module of the Elektronika S5-21 microcomputer has 1.5 times more lines than the corresponding module of the Elektronika S5-12 microcomputer, and the lines of this module and the machine itself can be switched easily from input to output and vice versa), and also switching to the microcomputer various functions which were realized on TTL [transistor-transistor logic] circuits in the SF-29 instrument because the Elektronika S5-12 microcomputer was not fast enough. These functions involve controlling the step motors of the instrument. The increased speed of the Elektronika S5-21 microcomputer also makes it possible in certain cases to reduce the volume of programs.

A spectrum analyzer on a single-board Elektronika S5-12 microcomputer. This is a measurement instrument, in the particular case designed for harmonic investigation of electrical signals or signals being converted to electrical ones [36]. The microcomputer, operating in parallel on 47 frequency channels does the following: feeding of information from the measurement part; outputting information to the video monitor; weighing frequency spectra for each frequency channel; averaging measurement results in each channel for n measurements; computing the spectral density of output; peak and exponential averaging.

This application of the microcomputer is also interesting because it is fairly universal and the microcomputer in whose ROM memory these problems have been realized can be used as part of various instruments and systems which require the entire set of programs or some of them. Manufacturing such microcomputers is very convenient for the supplier enterprise because it does not require them to fabricate new types of ROM LSIC's, and for the customer because it makes it possible to sharply reduce the time required to develop equipment by using an already developed assembly component. Those applications as facilities for preparing the sphere of utilization of single-chip microcomputers are especially important because they make it possible to turn these devices into universal LSIC's of fairly broad application.

The search for these applications is being carried on by a number of scientific collectives. Proposals have already been made today to equip microcomputers as finished articles with programs for statistical processing of measurement results and algorithms for fast Fourier transforms.

The chromatographic information processing device built on the basis of the Elektronika S5-02 microcomputer is an example of the use of microcomputers in analytic instrument making [37, 38].

Chromatographs are widely used to analyze compounds and identify components of complex compounds in the chemical, oil refining, and gas industries and also in the key problem of environmental monitoring. Chromatography is traditionally a field of instrument making where computing procedures are considered an inseparable part of the measurement process and are done by specialized computers.

Attempts have also been made to interconnect chromatographs with general-purpose computers. However, either they failed to fully realize the required computing and logical functions or they resulted in the development of devices that were too expensive and cumbersome and not reliable enough. The use of microcomputers made it possible to eliminate these shortcomings. The Elektronika S5-02 microcomputer processes data received simultaneously from five chromatographs. While processing the chromatograms the microcomputer determines the zones of integration (beginning and end of the peak), the area of the peak, the time of arrival of the apex of the peak, the concentration of components in the mixture, and so on.

In such measuring devices, whose modes of use are quite complex and varied, the idea of using the microcomputer as a time-program unit which carries on the experiment according to a definite program selected by the operator from a set of programs stored in the ROM memory or according to a program put together by the operator during preparation for measurement and the experiment, is very useful. To do this it is desirable for the device to realize a language that makes it possible to describe the program of the experiment in conventional form. It is natural that when a microcomputer is built into the unit, the construction of such a language is entirely feasible.

The economic impact of the use of a microcomputer in such a device comes first of all from reducing data processing time by several orders and, consequently, from the possibility of using this information on a real-time scale, directly in the course of the industrial process.

Chromatographs are not the only kind of analytic instruments that need fast data processing and automation of the experimental process by means of microcomputer. Other such instruments are roentgen quantometers, diffractometers, microanalyzers, and the like.

The further reduction of microcomputer cost, the development of series production of single-board models of microcomputers and MFM's for them, and the transition to single-chip microcomputers spread all these advantages not only to complex measuring units and instruments, but also to simpler measuring devices that are widely used.

5.5. Medicine

Improving the precision and objectivity of testing and processing the results of analyses and measurements quickly are key challenges in modern medicine and meeting them involves broad use of microcomputers. The low cost and large-scale use of microcomputers make it possible to convert various articles and systems for data processing from unique models accessible today only to the largest research centers to mass-produced inexpensive articles that are simple to use and accessible to any medical institution.

The application of microcomputers in laboratory testing equipment has much in common with other classes of measurement instruments considered above. The principal goals that can be achieved here are: an increase in the labor productivity of the laboratory worker; simplification and reduction in the cost of instruments; the possibility of instruments working in a system, above all under conditions of clinical diagnosis laboratory centers which do mass preventive examinations of the population.

Let us look at several examples of medical equipment built on the basis of Elektronika S5 microcomputers.

The photoelectric densitometer for electrophoresis, built on the basis of the Elektronika S5-11 microcomputer supplemented with an alphanumeric printer, performs the following tasks: conversion of a voltage proportional to the magnitude of light transmission; taking logorithms; integration of the electrophorogram along the entire length and in the segments bounded by minimums; computation of the concentration of components of the electrophorogram; outputting results to the display and the printer.

The calorimeter of serial laboratory analyses, based on the Elektronika S5-11 microcomputer and an alphanumeric printer module, performs these functions: conversion of a voltage proportional to the magnitude of light transmission; taking logorithms; constructing and periodic correction of the calibration curve; calibration of measurements; outputting results with an indication of the "norm-pathology" boundary and symbols to identify the specimen with results.

Expanding the number of problems solved by the calorimeter in a biochemical autoanalyzer makes it possible to insure programmed control of blocks that feed cassettes containing biospecimens, precise dosing of chemicals; and, synchronous control of thermostat, shifting, and transporting units [39].

The automated observation system for the condition of cardiology patients [4], constructed on the basis of Elektronika S5-11 and Elektronika S5-01 microcomputers, is a hierarchical system consisting of cardiomonitors at the bedside and a central position.

The bedside monitor, which can work not only within the system but also independently, is figured to serve one patient. Its microcomputer performs initial processing, analysis of the electrocardiosignal, formation of the diagnosis and alert signals, storage of cardiosignals in the pathological sectors, detection of this signal and the condition of the patient, and transmission of data to the central position.

The central position is designed to monitor and display the current condition and history of development of the disease of eight patients, to predict a worsening of the condition of the patients, and to output report documents. In this case the Elektronika S5-01 microcomputer performs additional processing of the data received from the cardiomonitor taking account of information from clinical examination and analysis for a more precise diagnosis of the patient's condition, forms an evaluation of trends toward change in the condition of each patient, predicts a worsening of the condition of the patients, records information, displays it in conventional form, and permits the doctor to interact with the system.

There is no doubt of the importance of rapidly organizing series production of these systems and supplying them to as many hospitals as possible. It is beside the point to speak of the economic impact of these systems; they are built to control serious cardiac illnesses in human beings.

The biotechnical unitfor occupational selection is also a hierarchical system constructed with microcomputers of the Elektronika S5 family. The single-board model of the machine is the basis for constructing the work position of the test subject and carries out initial processing of the subject's reactions signals to the test assignments that are given as well as processing physiological data on the test subject. The multiboard Elektronika S5-02 model carries on secondary processing of evaluations received from the work positions, allows the instructor to interact with the computer system, and so on.

Without dwelling on the importance of supplying modern, automated, mass-produced and inexpensive equipment for the process of evaluating the professional suitability of persons for many specializations such as pilots, air traffic controllers, and energy system controllers, we will simply offer some information on the benefits that can be obtained by constructing such systems as decentralized microcomputer-based systems [41]. In comparison with a system based on an M-6000 computer, the microcomputer system provides: a savings of more than 100 square meters of space for a classroom with 20 work positions; a reduction in service personnel from 5-7 to two persons; a significant improvement in working reliability; and, an increase in the level of standardization from 70 to 90 percent. The microcomputer system costs, at a maximum, one-third of the cost of a similar complex constructed on "rigid logic."

5.6. Automation

The general-purpose digital regulator [42] is a simple device in which the use of an Elektronika S5-11 single-board microcomputer as the primary part gave the regulator a number of very important features.

In the first place, it has available a broad set of software correcting elements that realize the six most widespread laws of regulation: proportional; integral-differential; proportional-differential; aperiodic; proportional-integral; proportional-integral-differential. The possibility is also envisioned of switching from one law of regulation to another during the working process following an external signal, as well as the possibility of changing tuning coefficients within a broad range.

In the second place, it offers the possibility of using the regulator in different regimes and receiving the control action in either digital or analog form, with unipolar and multipolar signals, and the like.

In the third place, it has a very flexible procedure for emergency protection against overload and incompatibility, as well as procedures by which the regulator monitors itself during the working process and upon starting.

In the fourth place, it offers a simple link-up with a higher-level computer in hierarchical control systems.

Such a regulator was built for the first time as the lowest element in a multichannel system to control the loading of complex aviation design elements during the testing process. Several dozen such regulators work under the control of one minicomputer. After it was built, however, its suitability for controlling other types of testing machines became obvious. Later its application to an even broader class of objects for processing control inputs with frequencies up to two hertz became apparent.

The only restriction is the cost of such a regulator, which is 2,000-3,000 rubles. This is entirely feasible for complex objects where the above-listed benefits of the digital regulator are especially important. It must be kept in mind here that rapid growth in the degree of integration is already today making it possible to build such a regulator in the form of a single-board design which reduces the price several-fold, and the transition to single-chip microcomputers, and further to those which have analog-digital and digital-analog convertors on one chip, will make the general-purpose digital regulator unit a mass-produced article of the electronics industry available to construct practically all types of servodrives without a limitation based on the frequency of the control input.

For this reason, as already observed above, the algorithms of the general-purpose digital regulator are extremely interesting as an example of standardization of an algorithm which converts a microcomputer with programs in ROM memory from a special-order item to a universal element of electronics engineering.

A control system for a ventilation unit [43] is another example of a control system based on the Elektronika S5-11 microcomputer that is designed for technical processes that require maintaining a constant speed of airflow with changing values of the cross-section of the airflow and air temperature and pressure.

Unlike the analog and static regulators ordinarily used for these purposes, the microcomputer-based system feeds assigned and measured parameter values with the necessary accuracy, computes control actions with the necessary accuracy, outputs the values of the basic parameter being regulated in a ready-to-use form that does not require additional computation based on formulas, monitors the work of the equipment, analyzes emergency situations, and makes necessary decisions on the further work of the equipment.

The software that has been developed provides high working reliability for the unit as the result of monitoring and analyzing trouble and broad application of tabular methods of computation with interpolation of results to compensate for the comparatively low speed of the microcomputer.

The use of a microcomputer in this unit makes it possible to eliminate manual operations and raise productivity by 10 percent. Maintaining the assigned air flow speeds automatically and reducing the time required to switch from one regime to another provide a considerable energy savings.

5.7. Questions Relating to Organizing the Application of Microcomputers

The effect of the application of microcomputers in a particular field of engineering cannot be enjoyed simply by making correct technical and algorithmic decisions. As has been observed more than once already, a broad range of production and organizational questions must also be resolved.

The methodology of work on the application of microcomputers is worthy of independent study and description. We will treat only certain matters which we consider most important: selection of paramount applications; organization of interaction between the microcomputer supplier and customer; and, forming the collectives to develop equipment based on microcomputers.

The selection of priority applications is an extremely important task, because the success of the first development is a good guarantee that work will continue on the application of microprocessors, not only in the specific collective, but often also in the entire branch of engineering; by contrast, failure may discredit the very idea of application for many years. A large majority of the examples of microcomputer applications that we have considered can be acknowledged as successful not only because they went through the experimental stage quite quickly and supported the development of series-produced models of equipment with new characteristics, but also because they were followed by a broad assortment of microcomputer-based equipment. For example, the spectrophotometer we described was the forerunner of a series of different classes of spectrophotometers equipped with the Elektronika S-5 microcomputers and, most importantly, of a broad assortment of optical instruments in which the technical, algorithmic, and organizational experience obtained during development of the spectrophotometer was employed in one form or another.

So, what application should be chosen first? Of course, it should be the one that is efficient and can be done quickly. Evaluating the efficiency of microcomputer applications in different devices was discussed earlier, but we should discuss the question of speed separately. The concept of speed of application has two components: speed of development and speed of organizing series production. These components differ and the ratio between them will differ in different applications, so from a series of possible developments it is necessary to choose the one which can be brought to the stage of a prototype (or model) in 1-1.5 years, and the manufacturing plant for the article should be able to expend minimum resources for the transition to production of this article. This period of time for production of the prototype enables the technical leadership to evaluate the advantages of the new article quickly, while the effect of producing the new article with minimum expenditures will interest the production services in further introduction of this type of article. This approach ultimately insures the earliest possible beginning to repayment of those expenditures which the state has incurred to develop the new generation of computer equipment.

In view of this, we will try to determine what form of development based on microcomputers can insure the necessary speed of realization of the expected impact.

The following classifications can be used for the types of development that employ microcomputers:

1. Modification of a system now being produced (setting up a new system) or particular elements of it, in which all computing facilities or part of them are replaced by microcomputers;

2. Developing, on the basis of an existing article, new equipment in which new essential functions are realized by means of microcomputers;

3. Switching existing mechanical, electromechanical, or electronic parts of series produced instruments that perform data control, processing, or transmission functions to microcomputers;

Each of these types of development has its own strengths and weaknesses when selected as the priority microcomputer application in equipment, an instrument, or a system. Let us consider each of them from the standpoint of design time, training cadres, efficiency of results, and the speed at which series production of the microcomputer-based article begins (see Table 5.1 below), always remembering that the enormous economic impact which this new generation of microprocessor technology can produce will be received only when series production and operation of this equipment begins. As the table shows, the first form of development is inferior to the other two, despite the positive value of one of the decisive factors in successful development — the technical readiness of specialists to carry it out — because of the great length of time between the beginning of development and realization of the impact of the use of microcomputers, the relative complexity of this type of development itself, and the multiplicity of technical questions that must be answered in the stage of introducing DCPT systems in series. And the long wait for the (first) positive results in this new area — designing DCPT systems based on microcomputers — certainly does not have a very good effect on the attitude toward this unquestionably progressive method of development and toward the participants in it.

Even considering the fact that in most cases the programming on the microcomputer must be done by the equipment developers themselves, who are specialist in electronics engineering (often working together with engineer-programmers, which we feel is the best combination), the second type of microcomputer-based development is much preferable to the first if only because the volume of specific software is usually less than in the first case and the positive impact of the use of microcomputers becomes apparent in the stage of the working model. But in this case the costs, in time and effort expended to put a new, with respect to design and function, instrument into series production at the manufacturing plant and to incorporate its use by the customer should not be forgotten.

In view of this, the third type of development is the best. While its advantages are just as apparent, it has the shortest realization time from the idea to receiving a practical impact because changes in the article already produced are minimal for the manufacturing enterprise, the customer has already mastered its sphere of application, the scope of production is fixed, and there are other beneficial aspects that usually accompany a series-produced article. The success of this type of development depends entirely on the designers of the article and does not involve questions that usually arise during new development such as supplying data sensors and transmitters, organizing communications lines, selecting a new configuration of assembly components, changing fixtures, and the like.

Organizing interaction between the microcomputer supplier and customer is far from a trivial matter. During the development of the Elektronika S5 family of microcomputers several organizational-methodological principles were worked out and employed to build the equipment and insure interaction between the microcomputer customer and the supplier. These were made the basis of the algorithms for work to apply general-purpose microcomputers (see Figure 5.8 below).

This algorithm makes it possible to partially overcome the conflicts between the short periods of incorporation of microcomputers with an assigned program in series production and the long periods required to develop and incorporate series

Type of Development Based on Microcomputers	Efficiency of Results	Cadre Training	Design and Development Time	Time of Beginning of Series Production
Replacement of Existing Computer Equipment with Microcomputers	Difficult to give a real quantitative accounting. Appears in fact as improved quantitative features of the object as a whole after system is fully developed.	Not Needed	Maximum because of: (1) complexity of systems and debugging facilities for them; (2) dependence of debugging on work of the object; (3) comparatively large volume of software.	Delayed because of the need to fully develop the entire system and see that all computing facilities, sensors, communications line equipment, and often the object itself are ready for series production.
Development of functionally new article	Increases productivity of operator labor. Increases the quality of the information processed. Takes load off data transmission channels, and etc.	Needed	Medium because of: (1) complexity of debugging the algorithm to realize new functions; (2) need for major design modification of article; (3) complexity of interlinkage with old element base.	Delayed because of questions that arise during production of the new article at the manufacturing plant.
Replacement of rigid logic with microcomputer	Lowers equipment cost and labor-intensiveness of its manufacture. Reduces weight and power consumption. Makes possible production without introducing additional capacities, and etc.	Needed	Minimal because of: (1) limited nature of functions (1-2K words); (2) simplicity of debugging facilities.	Determined entirely by the incentive of the manufacturing plant to produce the modernized article.

Table 5.1

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

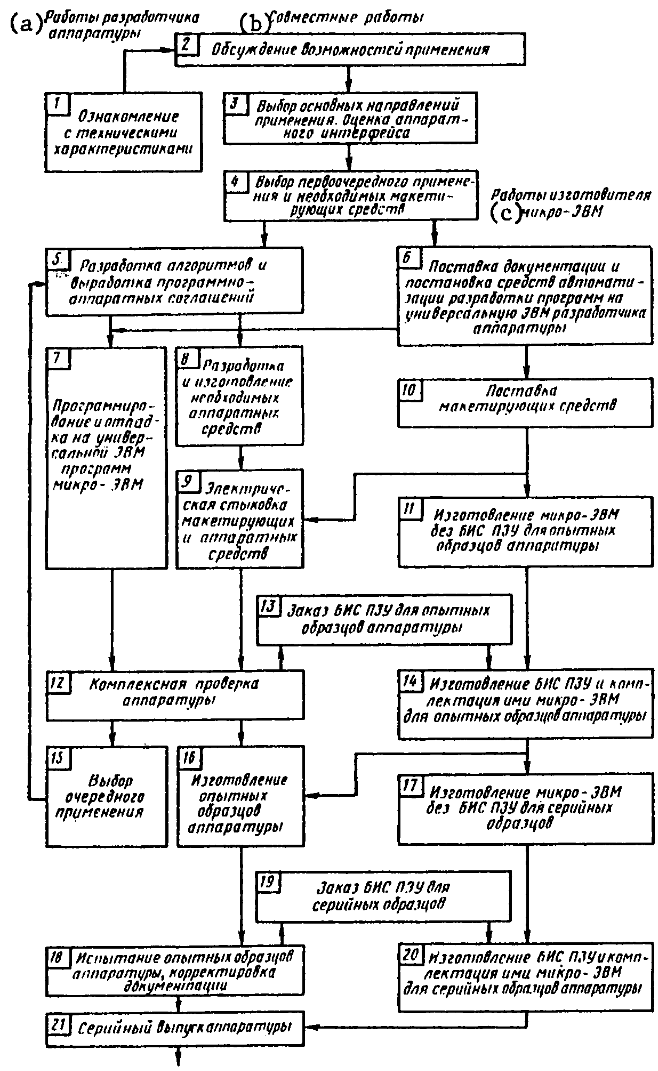


Figure 5.8. Algorithm of Work To Apply a Microcomputer.

Key:
(a) Work of Equipment Developer;
(b) Joint Work;
(c) Work of Microcomputer Manufacturer;
(1) Familiarization with Technical Specifications
(2) Discussion of Possibilities of Application;

[Key continued next page]

[Key to Figure 5.8 continued]

(3) Choice of Primary Areas of Application, Evaluation of Equipment Interface;
(4) Selection of Priority Application and Essential Modeling Facilities;
(5) Development of Algorithms and Working Out Hardware-Software Compatibilities;
(6) Delivery of Documents and Setting Up Facilities To Automate Program Development on the General-Purpose Computer of the Equipment Developer;
(7) Programming and Debugging of Microcomputer Programs on General-Purpose Computer;
(8) Development and Manufacture of Necessary Hardware;
(9) Electrical Interface of Modeling and Hardware Facilities;
(10) Delivery of Modeling Facilities;
(11) Manufacture of Microcomputer Without ROM LSIC's for Equipment Prototypes;
(12) Full Testing of Equipment;
(13) Ordering ROM LSIC's for Equipment Prototypes;
(14) Fabrication of ROM LSIC's and Supplying Them to Microcomputers for Equipment Prototypes;
(15) Selection of Next Application;
(16) Manufacture of Equipment Prototypes;
(17) Manufacture of Microcomputers Without ROM LSIC's for Series Samples;
(18) Testing Equipment Prototypes, Correction of Documents;
(19) Ordering ROM LSIC's for Series Samples;
(20) Fabrication of ROM LSIC's and Supplying Them to Microcomputers for Series Samples of the Equipment;
(21) Series Production of the Equipment.

---

production of systems for data control and processing based on them. This is true thanks to the following measures:

a. Purposeful and coordinated selection of primary and priority microcomputer applications (Stages 1-4, Figure 5.8);

b. The use of standard software for automating program development, produced on general-purpose computers for microcomputers, and the use of modeling facilities (Stages 6, 10);

c. Combining the production of microcomputers without ROM LSIC's for equipment prototypes with the stages of design and development of models of the equipment (Stages 5, 7-9, 11, and 12);

d. Combining the production of microcomputers without ROM LSIC's for series samples of the equipment with the stages of testing its prototype (Stages 16, 18 and 14, and 17).

Stages 1 and 2 (familiarization with technical specifications and discussion of the possibilities of using microcomputers) largely determine the decision to use the particular type (family) of microcomputers in the proposed equipment. During these stages the fundamental possibility of using the particular family of microcomputers and the extent to which the efforts of the equipment developers and the interests of the microcomputer supplier coincide are determined.

Interests of the Equipment Developer	Interests of the Microcomputer Supplier
Receiving essential auxiliary facilities, help in incorporating new methods of design, concrete help in development (above all software), guarantees of delivery of computer equipment with the required specifications	Rapid introducibility of development (above all the priority development), efficiency of use of microcomputers in equipment, stability of annual consumption of a sufficient number of similar (with respect to ROM LSIC's) microcomputers

The final document of these stages is the protocol (contract, decision), which defines the following: the prospects for use of microcomputers in the particular class of equipment; the list of basic areas of application; the stages and times for conducting priority development work to apply microcomputers; mutual guarantees of approximate deliveries and consumption of microcomputers for periods of five and seven years; times for supplying equipment developers with the necessary documentation on the microcomputers; times for setting up facilities to automate the development of microcomputer programs on general-purpose computers; times for setting up modeling facilities for comprehensive development and stand testing of the equipment developed.

A questionnaire survey of the expected microcomputer user is carried out in Stage 2 to evaluate the information value of the documents distributed concerning the microcomputer, to identify the most "attractive" (the term "attractive" should be understood in the broad sense, referring to those spheres of microcomputer application which provide the maximum national economic impact) spheres of microcomputer application from the standpoint of the equipment developer, to receive an idea of the level of readiness of the equipment developers to employ microcomputers, to clarify the directions of development of ways to step up equipment development based on microcomputers, and to analyze the suitability of current and future microcomputer characteristics.

Stage 3 includes selection of the basic applications and evaluation of the set of equipment. On the basis of a reconciled list of basic areas of application a list of equipment to which the application of microcomputers is unquestionably efficient is composed. It is considered here that the first and third types of development usually provide a reduction in cost, dimensions, and power consumption of the equipment, while the second increases the productivity of the apparatus itself or of the man-machine system. These advantages always occur when the necessary series production of microcomputers is guaranteed.

During the process of working out this list for each type of equipment there is a preliminary assessment of the hardware-software division of functions, that is, a determination of the basic functions assigned to the microcomputer and the functions which for some reason or other are to be kept in the old element base. This assessment makes it possible to clarify the configuration of equipment for interface with the microcomputer.

A contract on delivery of the necessary documentation concerning SARP [facilities for automating program development] and microcomputer hardware and putting the microcomputer cross-SARP on the general-purpose computer most accessible to the equipment developer is concluded.

It is very important in this stage to determine the nucleus of the collective of equipment developers and to begin training them to use microcomputer-based design software and hardware.

Stage 4 involves selection of the priority application and choosing essential modeling facilities. We have already considered this above.

At the completion of Stage 4 the enterprises that are manufacturing the microcomputer and the equipment developer compose a document that specifies the assortment of chosen types of modeling facilities to be delivered (or leased), the time when the equipment developer enterprise will provide information for the fabrication of ROM LSIC's, the times for delivery of microcomputers with ROM LSIC's and their number for an experimental batch of equipment, and the times for the beginning of series deliveries of microcomputers with an approximate number for the first two years.

Stage 6 includes the delivery of documents and setting up the SARP on the general-purpose computer. The enterprise that is manufacturing the microcomputers turns over to the customer the documents necessary to realize the priority and prospective applications according to a list agreed upon in the stage where the contract was concluded on delivery of documentation and setting up the SARP on the general-purpose computer.

Setting up the SARP on the general-purpose computer of the equipment developing enterprise is a similar "capital investment." In this case the enterprise that is manufacturing the microcomputers takes over the functions of training the first group of specialists in writing programs for the microcomputer, consults on the use of software and (where necessary) supplementing it in relation to the particular sphere of application, and the functions of escorting the SARP.

Practical design of the microcomputer-based equipment begins with Stage 5, the writing of algorithms and coordination of the interface (see Figure 5.8). The main job in this stage is to divide up the control contour or element of it for realization at the program (microcomputer) or hardware (interface) level. The most important criteria that determine the boundaries of this division for the given functions are the speed parameters of the microcomputer being used and the economic efficiency of realizing the function at the particular level. Establishing this boundary is in essence formulating the requirements for the hardware and software of the particular equipment to make it possible to program and debug specific programs on the general-purpose computer (Stage 7) parallel with development and manufacture of models of the equipment (Stage 8) and interconnecting models of the equipment with modeling facilities (Stage 9).

For the developers of the microcomputer-based equipment the comprehensive system testing (Stage 12) completes development of the ideology of application, making it possible to carry all its decisions (at the program level) to full satisfaction of the assigned specifications without any redesigning (and the related production questions of modification of boards, wiring, and the like). Stage 12 should conclude with functional testing of the system conducted by the developer together with the primary user and the manufacturing plant, recording of data that insures fulfillment of these tests, and supplying this information to the supplier of the microcomputer for the fabrication of ROM LSIC's (Stage 13). It is wise to begin a new cycle of equipment development in this period (Stage 15).

Stages 16, 18, and 21 are carried out by the equipment developer according to normative materials adopted in the particular sector. The microcomputer for the equipment may be delivered well ahead of existing schedules for building experimental and series-produced samples of the equipment (Figures 14 and 17) thanks to the advance manufacture of microcomputers without ROM LSIC's and supplying them to microcomputers as it becomes necessary to carry out the corresponding deliveries (Stages 19 and 20). This circumstance further reduces the cycle from development to introduction by improving the organization of Stages 16, 18, and 21 and reducing the time they require [44].

The question of forming the collective for the first microcomputer application is especially timely for an enterprise which is first encountering the problem of introducing computers directly in the sphere of data control, processing, and transmission. The complexity of this matter is reduced only slightly even when the enterprise has specialists in the operation of general-purpose computers and the formulation and solution to calculation and economic problems. The approaches to these jobs are different in many ways.

Most of the examples of applications of Elektronika S5 computers that we have considered were developed by collectives which had not done prior work on setting up computer-based systems and questions of programming. They consisted of specialists in designing the equipment which was to be switched to a microprocessor footing.

As the first years of work and experience developing several dozen types of equipment on a microcomputer base demonstrated, the minimum size of the group of specialists to develop the first experimental prototype of a piece of equipment whose complexity is evaluated from the standpoint of application of microcomputers in it by a program volume of 1,000-2,000 words is three persons.

A microcomputer with this memory volume can replace an electronic block containing 400-500 series 155 integrated circuits.

The labor-intensiveness of developing circuits to interlink the microcomputer with the other parts of the instrument and to develop the programs of its work are considered here; all the specific work of designing the instrument itself, manufacture, and independent adjustment (to the point of interlinking with the microcomputer) is not taken into account. This collective can develop an experimental prototype on the basis of an existing model of the controlled equipment in 10-14 months, including time for retraining specialists.

Who are the best members to make up such a group, and what functions should the specialists in it have?

One of the members (usually the project leader) should be a specialist in the design of the particular type of equipment into which the microcomputer is being built. It is absolutely essential that this specialist have a detailed knowledge of the characteristics of microcomputers as facilities for information control and processing and it is not necessary at all for him to be a specialist in computer structure and the subtleties of programming. He will receive the necessary knowledge in these matters during the development process.

It is wise for one of the other group members to be a trained electronics specialist capable of studying the microcomputer at the structural circuitry level. This person should have a detailed understanding of the power supply system and output circuits of the microcomputer, its timer circuits, and the interrupt system; a general idea of programming technique is also necessary. The second member should be a programmer who has already mastered all the procedures of programming in machine code and autocode, can handle the scaling procedure for work with a fixed-point computer, and has studied the microcomputer interface (but not the parameters of the signals being interconnected!).

Both these specialists should understand the characteristic features of the work of the equipment that is to be automated and the technical objectives which should be accomplished by use of the microcomputer. The simplest way to train these specialists is to select them from the developers of traditional electronic assemblies and blocks or automation systems for the particular class of equipment. Experience shows that it is simpler and easier to train these engineers to be microcomputer electronics specialists and engineer-programmers than to "introduce" a professional programmer and computer designer into the collective that has developed the equipment on the basis of microcomputers.

How should specialists be retrained to do this job? The simplest, but not the fastest way, is independent study of the foundations of computer technology and programming and then mastering the particular model of microcomputer. The many books in this area will unquestionably help. The authors hope that the present book will also make this training somewhat easier. Another way is the schools for raising the qualifications (advanced training) of engineers. Higher educational institutions have many such schools. Many of them have already taken on the exceptionally important job of rapidly teaching the technology of applying microcomputers to a very broad range of specialists in electronics, automation, and programming who work in various sectors of the economy. It is particularly efficient to organize this training on the sectorial principle because it is very important to combine study of the material base of the sector while making specialists aware of the economic and organizational aspects of the application of microcomputers in the particular sector.

For this reason, institutions of higher technical learning should make the study of microprocessor technology and programming an inseparable part of the training of specialists not only in computer technology, but also in the fields of automation, measurement technology, radioelectronics, and radio measurement. The fundamentals of microprocessors and the capabilities which they offer should be studied also by future "consumers" of microcomputer-based equipment: production engineers, machine builders, and others so that they will later be the moving force in formulating demands for the control elements of equipment and developing fundamentally new types of equipment based on microprocessors. After all, the most promising areas of microprocessor application are those which have not yet even been discovered.

Training cadres is an important aspect of the overall problem of rapid, widespread introduction of microprocessers in the economy and a timely solution to this problem should be a focus of attention for leaders at all levels.

Conclusion

It would be advisable here to review certain aspects of the technology of producing LSIC's which are closely linked to the development and application of microcomputers.

The first aspect is the maximum feasible degree of integration of elements on one chip. If we limit ourselves to a consideration of MOS technology, in 1979 memory circuits with 65,000 K words and up to 300,000 transistors in regular structure on a chip were developed, as well as single-chip microcomputers with 50,000-100,000 transistors in irregular structure on the chip. The minimum geometric dimension of these elements is 3-4 microns. The possibilities of planar technology have not been exhausted today. In the near future it is clear that the minimum dimensions of the elements will be lowered to 1-1.5 microns. Along with the reduction in the size of elements there is a trend to increase the size of the chip as optical mechanical equipment becomes more sophisticated.

This means that in the near future the number of transistors on a chip will increase by at least one order, and the speed of LSIC's will also increase because of the decrease in the length of the channel and the reduction in the volume of the circuits themselves. This raises a basic question: how should these new capabilities be used from the standpoint of building microcomputers? There are two possible answers. The first is to develop a general-purpose microcomputer on a single chip that has all types of semiconductor memory (main memory, ROM memory, and semipermanent ROM memory), all types of input-output channels (digital parallel and sequential, and analog channels of different speeds), and linkage units with terminal devices on one chip. The second way is to develop specialized units with the most efficient use of the capabilities offered by contemporary semiconductor technology for each particular application.

The shortcoming of the first solution is that it does not make full use of the capabilities of the chip for the specific type of application; the difficulty with the second solution is that it overbroadens the assortment of microcomputer LSIC's, which in this stage of the semiconductor industry can lead to the problem of a broad assortment of special-ordered LSIC's, which already existed before the age of microprocessors. A compromise solution is evidently needed: develop fairly universal devices which allow the most diverse applications, and create a limited number of specialized models of microcomputers for the most common technically and economically sound applications. Therefore, consideration of the spheres of application of microprocessors and microcomputers is a key issue in the development of this area of microelectronics.

Of course, at the new level of LSIC technology there will be not just single-chip microcomputers but also fairly powerful functional units to build new generations of computer facilities, including single-board models of the next level and more complex units. It is important in this case to emphasize the use of completed units, which the microcomputers are, as the most widespread means of automation.

Because the development of microcomputers and organization of their production is not a goal-in-itself, but rather a means to automating a large number of instruments and devices used in the national economy, we should caution once more against the possibility of widespread introduction of microprocessor technology in the national economy lagging behind the growth rate of production of microprocessors and microcomputers and the rate of development of new, more sophisticated models of microprocessor technology. Therefore, we should also direct attention to one of the most productive methods of reducing the cycle from development of the new basic element to introduction of equipment based on it: the method of designing equipment after ratification of specifications for new microelectronic units parallel with their development.

The principal prerequisite for successfully doing this is continuity of the generations of microelectronic devices, which is secured by preserving all the basic characteristics of the microcomputer family as their technical-economic indicators are improved. This book has demonstrated this proposition in detail. We need only add that the very first model of a single-chip microcomputer in the Elektronika S5 family corresponds fully with respect to functional capabilities to the single-board Elektronika S5-11 microcomputer, while significantly surpassing it in speed. This approach by the developers and producers of microcomputers makes it easier to solve the problems of organizing application, speeding up the beginning of use of articles based on the technology by 2-3 years.

The second aspect of the technology of LSIC production is the economically acceptable volume of production of microprocessors and microcomputers.

Group methods of technology used in planar technology presuppose simultaneous fabrication of a set of articles shaped on a silicon plate. The number of uusable articles per plate in addition to the percentage of output of usable ones (the eternal problem of production engineers and developers) is determined by the size of a chip and the area of the silicon plate. The diameter of the plates being processed is increasing from 40-60 to 75-100 millimeters with a trend toward a further increase. Despite a certain increase in the size of the chip, the output of usable chips per plate increases significantly through an increase in the diameter of the plates that are processed. Thus, increasing the complexity and functional completeness of the articles is accompanied by growth in their series production.

The development of articles with submicron geometric dimensions to the elements is a further stage in the development of LSIC technology. Electronic and roentgen lithography are the basis of this technology. Without going into a discussion of the problems of this stage, which are related both to the technology of achieving such dimensions and the physical limitations of supersmall structures, the authors emphasize once again the consistent development of the technology of superlarge integrated circuits and, as a result, the appearance of new problems in the application of these articles. And if the readers of the book, along with development of the application of models of microcomputer series already developed, look into the future use of microcomputers in the national economy the authors will consider their job done.

11,176
CSO: 1863/188

SOFTWARE

UDC 51 : 681.3.06

TEXT EDITOR WITH DEVELOPED SYSTEM OF COMMANDS FOR YES OS

Moscow PROGRAMMIROVANIYE in Russian No 3, May-Jun 81 (manuscript received 7 Jul 80) pp 39-48

[Article by N. N. Bezrukov, Kiev]

[Excerpt] The editor has been used since August 1979 and experience has shown that the user performs editing much faster with the help of NEATED than with the aid of the standard YeS OS editors. Although the NEATED command system is rather large, it is mastered by the majority of users within about a month. Also, the user need not master all commands right away: he can restrict himself to a subset (for example, addressing by using the logical numbers of the lines and the insert-delete commands without the update modifiers). The other commands may be mastered as the need arises.

Among the shortcomings of NEATED is the large amount of storage needed for its operation (on the order of 80K) which is almost double that needed for simpler editors. However, this shortcoming is becoming less perceptible in connection with the transition to computers with a main storage size of several Mbt. At the same time, user comments indicate that practically any transformation of text may be performed by using NEATED (although not always simply). NEATED has been used to convert programs from FORTRAN-IV to PL/1.

NEATED is included in the NEAT program package [14-16] developed at the UkSSR Scientific Research Institute for Psychology.

8545
CSO: 1863/226

UDC 681.3.014.001.63

TOOLS FOR MODELING REAL-TIME SYSTEMS FOR THE 'EL'BRUS' MULTIPROCESSOR COMPUTER

Moscow PROGRAMMIROVANIYE in Russian No 3, May-Jun 81 (manuscript received 18 Feb 80) pp 49-54

[Article by K. G. Suleymanov, Gomel']

[Text] Program tools for designing real-time programs for the "El'brus" MVK [multiprocessor computer complex] are described in this article. Programs are designed in the specially developed Opal language. The tools have been implemented on the BESM-6 computer and consist of an Opal translator and a model of the "El'brus" MVK which includes a model of OS.

1. Prerequisites for Development of Tools

Described in this article are the program tools developed on the BESM-6 computer with OS DISPAK and designed to be used in developing real-time programs (RV-programs) implementable on the "El'brus" multiprocessor computer complex (MVK) [1, 2]. Development of these tools was necessary for the following reasons:
a) there was no "El'brus" machine at the time tool development began. The available "El'brus" interpreters on the instrumental computers could not satisfy the real-time program developers, since they could not interpret the "El'brus" operating system and multiprocessor capability [3]. With these interpreters, it was not possible to execute multiprocessor jobs for working off the interaction of the processes of the jobs between themselves and with the operating system; and
b) the traditional method used in designing complicated systems is (program) simulation. However, use of this method in designing real-time programs entails certain difficulties.

The languages usually used for simulation most often do not coincide with the language for implementation of real-time programs. Therefore, in the simulations it is not possible to adequately reflect the static structure of the text of real-time programs. Consequently, there remain unresolved questions of structuring and ensuring correctness [4-6] that are of paramount importance in developing real-time programs, considering their large size and complexity.

Great difficulties arise in organizing joint execution of a model of a specific computer and a model of a real-time program within the bounds of one simulation language which prevents adequate reflection of the dynamics of functioning of the real-time program. Questions of development of the additional OS layer oriented to operation in real-time are unresolved;

c) control and analysis algorithms that support the required functioning of objects of the external environment and realized in the real-time program do not remain unchanged throughout the development phase. In addition, development of some control algorithms may outstrip development of other algorithms, while certain algorithms may be fully defined only at the very end of the development phase. All this creates a very unstable situation in programming and requires creation of corresponding instrumental software allowing the developer to make operative changes so that as a whole the real-time program remains clear and easy to analyze.

Within this article, a real-time system (SRV) is considered the aggregate of the hardware, the computer operating system and the real-time programs. The principle difference between a real-time and a conventional program is as follows.

During execution on the multiprocessor computer complex, the real-time program synchronizes its actions with the events occurring in the outside world in "real time." Execution of conventional jobs occurs in "virtual time." Of course, this does not mean that there are no external events with problems solved in "virtual time." Such events may be, for example, the end of an exchange with external storage, receipt of a message from the operator, etc. The composition of external events for "virtual-time" problems is rather limited and lends itself to systematization.

A fundamental feature of the external events in a real-time program is the impossibility of standardizing these events to charge the operating system with the functions of synchornization and bufferization of the information circulating between the external objects and the real-time program.

In connection with this, the need arises of transferring to the real-time program certain functions usually reserved for the OS. These functions may be: dynamic determination of process priorities; processing interrupts from external objects; determining critical time intervals for operation with certain special devices, etc.

A very important characteristic of a real-time program is that it requires large computing resources and is usually the only job functioning in the real-time system. Consequently, the logic of operation of the entire system is largely concentrated in this one job. This fact has the following important consequence. Usually the OS's implementing multiprogramming contain general-purpose algorithms to distribute computing resources. These algorithms achieve efficiency in use of computing resources "on the average." This is because under the conditions of limited resources and the inadequacy of information on the "behavior" of the jobs in the multiprogramming mix, the latter cannot effectively affect the procedure for allocation of resources.

In the real-time system, the operating system can more precisely consider the need for resources by the processes of the sole real-time job, and the traditional list OS job directives for reources may be expanded with so-called "prompting directives." These directives, essential for real time, may be instructions on changing the status of file residence, relative priorities of processes, etc.

To realize the prompting directives in a "special layer" of the OS nucleus, one has to develop, possibly, privileged procedures for organizing the corresponding reactions to these directives. It is evident that these procedures must interact harmoniously with the appropriate procedures of the OS nucleus.

2. Description and Method of Use of the Tools

It is assumed that refinement of the real-time system specifications, development of the real-time program structure and the possible adaptation of the OS to specific real-time applications are done on the base of a complex dynamic model. Usually in such models, a basic part is the real-time program model, while the characteristics of the machine and operating system are drawn in largely for working off time delays. In this work, a different approach is taken to build the complex dynamic model of the real-time system. Its basic part is the model of the machine and operating system. To adequately represent the real-time program in the complex dynamic model, the special language, Opal, is proposed for describing the behavior of the problems. This is due to the features of the "El'brus" MVK [7] in which the real-time program is implemented:
a) multiprocessor capability;
b) orientation to programming only in high-level languages; and
c) high dynamicity of allocation of computing resources supported by hardware ((tegovskaya) architecture) and the operating system.

In accordance with these features of "el'brus," the Opal language:
a) provides facilities for "splitting" the real-time program into parallel processes;
b) allows description of the structure of programs to be implemented in high-level languages with a block structure, in particular in "El'brus" Autocode [8]; and
c) offers language constructions for using the basic dynamic properties of the machine.

Opal syntax and semantics have been realized in a way that lets the developer concentrate only on those features of the real-time problem that affect the overall system operating situation.

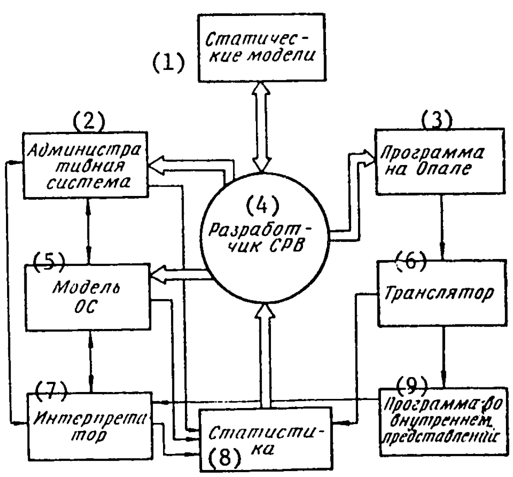
Opal's orientation to the "El'brus" MVK is expressed in that there are no fixed types in the descriptions, and information address type values have been introduced: indirect words, descriptors, procedure labels and semaphores. Value types are checked dynamically during interpretation. Interpretation of the Opal program allows concealing from the user system interrupts and transferring to the interpreter calculation of model time. Local data of procedures and blocks of the OPAL program are loaded into stacks just as data of the modeled procedures of OS.

In connection with the difficulty of modeling the main memory of the "El'brus" MVK on an instrumental machine, there is no capability of operating with values of files outside the stacks in the language. This means the language has no facilities for assigning a value to an individual element of a file outside the stacks and fetching this value. These facilities exist only for files in stacks, the sizes of which are limited to the stack sizes. The statement for operating with files outside the stacks, that is available in the language, models only time delays for processing these files in the user mode and operation of the storage control procedure in the system mode.

The language has statements to call procedures for interfacing the "El'brus" MVK operating system with the user. The interface procedures are the OS procedures included in the user context, i.e. can be started from user programs. Also included in the language are the usual facilities for modeling discrete systems: time delay statements and situation query statements.

The time delay statement allows simulating calculation of varying duration in the processes of the real-time problem in the course of which nor actions are performed that affect the operating situation. Situation query statements allow obtaining information on the effect of the behavior of the real-time program on the operating situation.

It is necessary to note that in developing the language there was no aim to make it as close as as possible to the "El'brus" Autocode. First, this would have caused too great an extension of development time. Second, it is very doubtful that within the bounds of the instrumental machine a model of the operating system and an interpreter of the full set of the "El'brus" MVK instructions could be realized jointly that is sufficiently convenient to operate. Third, the tools developed pursue a more modest (but very important!) goal: to help real-time program developers comprehend ways of mapping external process control algorithms to the "El'brus" MVK computing structures and the dynamics of interaction of real-time program processes between themselves and with the operating system, and thereby promote in the spirit of structured programming [9] a reduction in conceptual space separating the external process control problem, formed more or less precisely in terms of problem-oriented concepts, and its program realization on the "El'brus" MVK.



Key:
1. Static models
2. Administrative system
3. Program in OPAL
4. Real-time system developer
5. OS model
6. Translator
7. Interpreter
8. Statistics
9. Program in internal representation

The set of tools contains the following components: a) translator from the OPAL language in internal representation; b) interpreter of constructions of internal representation; c) model of the operating system; and d) the administrative system. A schematic representation of the organization of work using the set of tools is shown in the drawing.

The simulation cycle begins with the stage of so-called static simulation of the individual algorithms or groups of algorithms. Static simulation may be done in either "El'brus" Autocode or another high-level language. Static simulation results give the real-time program developer a more accurate notion of the problems solvable by the system and estimates (even if rough) of execution times and storage amounts required for the algorithms.

The information derived in static simulation is then used to map the algorithms to the "El'brus" MVK computer structures in the OPAL language.

Then the program in OPAL is translated into a program in internal representation which is an analog of the "El'brus" MVK instruction set. An OPAL development criterion was minimization of the number of constructions of the internal representation since this simplifies realization of the interpreter. The internal representation now consists of 43 constructions. (There are about 250 instructions in the "El'brus" MVK instruction set.)

After this, the program in internal representation is input to the model of the "El'brus" MVK which consists of the interpreter, the administrative system and the model of the operating system. The operating system model contains basically the models of the procedures for management of storage and processes. This is because the most critical computing resources in the real-time system are main storage and central processors.

All components of the set of tools have been realized in the Yarmo high-level language and connected as a component to the STRELA instrumental complex [10]. This complex has the status of a conventional user job in OS DISPAK. Thanks to this connection in operating with the set of tools one may use all the basic components of STRELA, including the archives and editor. Also, subsequent modifications of STRELA offered the capability of interactive operation with the set of tools.

The abundance of descriptive facilities in the Yarmo language and its nearness in this sense to "El'brus" Autocode simplified writing the models of the OS procedures, while the machine orientation of Yarmo made it possible to efficiently realize replacement of the stacks.

Then the statistical information derived in executing the program on the model of the "El'brus" MVK are processed and analyzed by the developer to detect errors and insufficiency of resources, etc. Insufficiency of resources may be taken care of by adding appropriate resources to the machine configuration or changing the algorithms. Errors in the program are eliminated directly in OPAL since the translator achieves unambiguous correspondence between the internal and OPAL representations of the program.

The corrected OPAL program is again input to the translator, thereby starting a new cycle of real-time system simulation.

3. Results Obtained

The OPAL program is a certain description of the block structure of the text of the parallel (multiprocess) real-time program. The level of detail of this description may be different for different parts of the real-time program and is selected by the developer as a function of the status of development, investigative aims set, etc. In any case, there must be detailed descriptions of the actions associated with generation and/or use of features essential to organization of the computing process on the "El'brus" MVK, including declarations of semaphores and semaphore primitives.

It is well known how complicated the parallel program can be that has interaction of processes organized on the base of the semaphore technique [11]. In studying the textual structure of the parallel real-time program, the developer may, for example in style [12, 13], prove the correctness of operation with common data (semaphores of critical intervals) and the correctness of event synchronization of processes ("private" semaphores of processes) and may construct program structures similar to conventional critical intervals and monitors [14-16] etc.

But for development of the real-time program, it is not enough to study just its structural aspects. Study of the dynamic aspects of the real-time program is also required for this. This study is done using the model of the "El'brus" MVK on which the OPAL program is executed. In the process, the following questions are resolved:

whether the strict time limits on control of the external objects are adhered to is determined;

the algorithms for planning priorities of processes are worked through;

the equipment configuration needed for realization of the real-time program is determined for the "El'brus" MVK (number of central processors and size of main storage); and

the possibility of using the mechanism of dynamic allocation of storage available in the OS for the "El'brus" MVK is determined. There may be three unpleasant cases here.

With unreasonable use of generators of local storage in procedures frequently being called, the mathematical storage of "El'brus," very large in size (2^{32} words), which is allocated to a problem without reuse, may in principle be exhausted. The overhead in time for operation of the OS procedures for storage management may be great. There may be an insufficient amount of physical main storage in "El'brus."

In the first two cases, careful use must be made of the language constructions leading to advancement of the indicator of free mathematical storage and to an increase in storage fragmentation. In the third case, the computational algorithm must be changed or the physical size of main storage in "El'brus" increased.

In operating with the set of tools, the mechanisms for the real-time program communication with external objects may be worked out preliminarily, in particular the scheme for processing interrupts from the external objects and the principles for organization of information exchange with external objects. The tools may also be used for training. For example, in studying the various aspects of interaction of the parallel processes in a multiprocessor machine.

* * *

The set of tools were checked in simulating various structures of program realization on the "El'brus" MVK of several practical real-time systems. The results obtained showed the practical usefulness of the tools in choosing and designing structures for real-time programs. If the size of the OPAL program does not exceed 4000 lines, then the entire set of tools fits within the user address space on the BESM-6 (32K words) and a rather fast interpretation of the OPAL program is achieved. For different real-time systems, a different slowdown was obtained--from 0.9 to 70-fold, i.e. in simulating one second of real time, approximately from one second to one minute of BESM-6 time was expended.

BIBLIOGRAPHY

1. Burtsev, V. S., "Tendentsii razvitiya vysokoproizvoditel'nykh sistem i mnogoprotsessornyye vychislitel'nyye kompleksy" [Trends in the Evolution of High-Throughput Systems and Multiprocessor Computer Complexes], Moscow, ITMiVT [Institute of Precision Mechanics and Computer Engineering], 1977.

2. Babyan, B. A., "Osnovnyye printsipy programmnogo obespecheniya MVK 'El'brus-1'" [Basic Principles of "El'brus-1" MVK Software], Preprint No 5, Moscow, ITMiVT, 1977.

3. Ivanov, A. P. and Semenikhin, S. V., "Operatsionnaya sistema MVK 'El'brus-1'" ["El'brus-1" MVK Operating System], Preprint No 2, Moscow, ITMiVT, 1977.

4. Wirth, N., "Toward a Discipline of Real-Time Programming," CACM, No 8, 1977.

5. Hansen, P. B., "A Comparison of Two Synchronizing Concepts," ACTA INFORMATICA, Vol 1, 1972, p 190.

6. Owicki, S. and Gries, D., "An Axiomatic Proof Technique for Parallel Programs I," ACTA INFORMATICA, Vol 6, 1976, p 319.

7. Pshenichnikov, L. Ye. and Sakhin, Yu. Kh., "Arkhitektura mnogoprotsessornogo vychislitel'nogo kompleksa 'El'brus'" [Architecture of the "El'brus" Multiprocessor Computer Complex], Preprint No 7, Moscow, ITMiVT, 1977.

8. Pentkovskiy, V. M., "Osnovnyye kharakteristiki Avtokoda MVK 'El'brus'" [Basic Characteristics of the "El'brus" MVK Autocode], Preprint No 6, Moscow, ITMiVT, 1977.

9. Dal, U.; Dijkstra, E.; and Hoare, C., "Structured Programming, Moscow, Mir, 1975.

10. Chinin, G. D. et al., "Complex of Software of an Instrumental Computer," in the book "Sistemnoye i teoreticheskoye programmirovaniye" [System and Theoretical Programming], Kishinev, 1974.

11. Courtois, P. J.; Heymans, F.; and Parnas, D. L., "Concurrent Control with 'Readers' and 'Writers'," CACM, No 10, 1971.

12. Dijkstra, E. W., "The Structure of the 'THE' Multiprogramming System," CACM, No 5, 1968.

13. Dijkstra, E., "Interaction of Sequential Processes," in the book "Programming Languages," Moscow, Mir, 1972.

14. Hansen, P. B., "Structured Multiprogramming, CACM, No 7, 1972.

15. Dijkstra, E. W., "Hierarchical Ordering of Sequential Processes," ACTA INFORMATICA, 1, 1971, p 115.

16. Hoare, C. A. R., "Monitors: An Operating System Structured Concept," CACM, No 10, 1974.

8545
CSO: 1863/226

UDC 681.3.06

REALIZATION OF PASCAL LANGUAGE FOR 'EL'BRUS' MULTIPROCESSOR COMPUTER COMPLEX

Moscow PROGRAMMIROVANIYE in Russian No 3, May-Jun 81 (manuscript received 13 Aug 79) pp 62-64

[Article by S. V. Vdovkin, A. A. Kubenskiy, S. S. Lavrov and V. O. Safonov, Leningrad]

[Text] Described are the source language of the PASCAL compiler for the "El'brus" multiprocessor computer complex and the general principles of realization and structure of the compiler. Some characteristics of the compiler and an evaluation of the realization are given.

In recent years, the algorithmic language PASCAL has become more and more popular [1]. The language contains convenient facilities for defining and processing structured data, including sequential files; it has the necessary set of control constructions and apparatus of procedures that allow parameters-variables and parameters-procedures. These facilities are specified in a simple, clear and uniform form, which makes PASCAL programs more intelligible and lends them special elegance. Another merit of the language is the strictness and compactness of the definition of its syntax. The language allows solving a broad class of problems, but it is most popular as an instrumental facility for developing system programs and as the basis for a study of programming [2].

An effort is underway in the computer center of Leningrad State University to implement PASCAL on certain computers.

A PASCAL compiler has been developed for the "El'brus" MVK [multiprocessor computer complex]; the compiler is intended for any class of users (students, system programmers, application program developers) and provides for obtaining an efficient working program, convenient facilities for debugging and correcting programs, output of information for the user with a varying degree of detail, and disengagement by the user of part of the dynamic checks to obtain a more efficient version of a program that has already been debugged.

The "El'brus" PASCAL compiler, just as any programming system for the "El'brus" MVK, is accessible in the computer environment which now includes part of the required capabilities: system of files for storage and editing of texts (programs, in particular), and a general-purpose interface for compilers that supports connection of modules in other languages, for recompilation and independent compilation of procedures.

1. Features of the Source Language. The authors of the realization strived to keep the source language as close as possible to that described in [1]. The only change in the syntax of the language, compared to the standard, is that in the construction "set," the type of elements is explicity specified.

On the other hand, to make fuller use of MVK capabilities, some expansions have been introduced:
the standard scalar types "short integer," "short real" and "long real" have been added;
in defining the file type, file attributes must be specified, otherwise they are established by default; and
in the selection statement, an alternative can be specified to be used when there is no alternative to the computed value of the selector.

In a number of cases where the treatment in [1] is not very clear, additional conventions were adopted; the most common are cited below.

a) The service symbols of the language, printed in [1] in boldface (begin, array, case, function, etc.), are graphically indistinguishable from identifiers, in accordance with the version of the language, which is admitted by its author himself [2]. Consequently, one may not describe in a program an identifier matching one of the special symbols.

b) Standard identifiers of types, procedures, functions, constants and variables, just as ordinary identifers, may be redescribed by the user.

c) Previous description of procedures and functions is not required. An identifier of a procedure (function) may be used prior to its definition.

In the rest of it, the language has been implemented without any restrictions, except natural quantitative ones which are rather few for the "El'brus" MVK (for example, identifiers are distinguished by the first 80 letters).

2. General Structure of the Compiler. The syntax of the language constructions presupposes, in the opinion of the authors, selection of one of the methods of top-down analysis and text scanning from left to right. In the majority of cases, the construction is recognized by the first lexeme: if--for a conditional statement, record--for type of record, etc. Only in certain cases is a look ahead by one lexeme required. Thus, global analysis cannot improve the quality of neutralization of errors, and the recursive descent method is used [4], combined in the necessary cases with the mechanism of passing through to the expected lexeme. It is possible that the more precise conformance to the syntax described in [1] has created the traditional difficulties associated with identification of names of procedures and functions. One-pass translation of a program, in which identifiers of procedures (functions) may be used prior to definition, requires construction of massive list structures for applicable occurrences of all global identifiers and inefficient repeated scans of these structures. Therefore, the decision was made to introduce a second pass as the most natural in which definitive identification is made. The following logic structure of the compiler and process of translation was adopted.

The compiler consists of blocks for the first and second passes. The first-pass block is formed by the subroutine for lexical analysis, procedures for processing descriptions and procedures for analysis of the statement division. The block receives for input the file containing the program text and the set of translation conditions. Output (and input for the second pass) are: table of identifiers, syntax graph of descriptions that contains all necessary information for both passes and code generation, set of local syntax graphs corresponding to the divisions of the statements of the program, procedures and functions, and a file of constants.

Performed on the first pass are complete lexical and syntactic analysis of the text, complete semantic analysis of the descriptions (with identification in the descriptions) and construction of the graphs of the descriptions and statements. In the statement divisions, identification is made only in "variable" constructions in the left parts of the assignment statements (since according to syntax, a function designator without parameters may not be encountered in them).

Errors detected in the first pass are localized to the smallest elementary construction (description, statement, expression). Regardless of errors, after completion of the first pass, the second pass block is started which consists of procedures for analysis of the internal representation and the program generator of codes. Performed are a trace of the tree of descriptions and the statement trees, complete identification, semantic analysis of statement divisions (in particular, semantic computation of types) and code generation. Code is generated only when there are no errors in the program.

3. Characteristics of the Translator and Evaluation of the Realization. As results of testing have shown, 8000 to 9000 usable instructions of the "El'brus-1" MVK are required for processing by the translator of one line of input text (which contains 30 to 40 characters, i.e. one to two descriptions or one to two statements) on the average. Since the average instruction execution rate on the "El'brus-1" MVK is 1,500,000 operations per second, the translation rate is 9000 to 12000 lines per minute. With these parameters, the translator is not inferior to, for example, the BESM-6 PASCAL system [5] (taking into account the difference in speed between the "El'brus-1" MVK and the BESM-6). The translator performs certain optimizations of object code, for example computation of constant expressions and calculation of the static components of addresses of variables during translation.

BIBLIOGRAPHY

1. Jensen, K. and Wirth, N., "Pascal User Manual and Report," 2d corrected ed., Springer Verlag, 1978.

2. Wirth, N., "Systematic Programming: An Introduction," Moscow, Mir, 1977.

3. Babayan, B. A. and Sakhin, Yu. Kh., "The 'El'brus' System," PROGRAMMIROVANIYE, No 6, 1980.

4. Bol'ye, L., "Methods of Constructing Compilers," in the book "Programming Languages," Moscow, Mir, 1972.

5. Pirin, S. I., "Implementation of the PASCAL Compiler for the BESM-6 Computer," in the book "Obrabotka simvol'noy informatsii" [Processing of Symbolic Information], issue 4, Moscow, VTs AN SSSR [USSR Academy of Sciences Computer Center], 1978.

8545
CSO: 1863/226

ABSTRACTS FROM THE JOURNAL 'PROGRAMMING', MAY-JUNE 1981

Moscow PROGRAMMIROVANIYE in Russian No 3, May-Jun 81 pp 95-96

UDC 681.142.2

AN APPROACH TO CONSTRUCTING A GENERAL-PURPOSE SCHEME OF LANGUAGE. SEMANTICS

[Abstract of article by V. A. Tuzov, Leningrad]

[Text] A version of a general-purpose scheme of language that may serve as a means of describing the semantics of programming languages is decribed in terms of compositions. The essence of the scheme is demonstrated in examples. Bibl. of 9 titles.

UDC 518.5:681.142.2

SUBSTANTIATION OF ALGORITHMS FOR CONVERSION OF LARGE-BLOCK PROGRAMS

[Abstract of article by V. N. Kas'yanov, Novosibirsk]

[Text] The author introduces a class of statement schemes with storage that are models of programs, the elementary statements of which may have an internal structure and operate with composite informational objects. Bibl. of 7 titles, 2 figures.

UDC 681.3.06

COMPUTER GRAPHICS IN PROGRAM BANKS

[Abstract of article by V. P. Shampal, Tallinn]

[Text] Discussed in the article is an approach to equipping program banks (BP) with computer graphic facilities (SMG). The suggested approach is characterized by the presence of high-level graphic facilities. Computer graphic facilities for program banks have been realized by using the YeS PRIZ instrumental programming system. Bibl. of 12 titles, 4 figures.

UDC 681.142

A METHOD OF PLANNING OF CALCULATION CHAINS

[Abstract of article by Yu. A. Bukhshtab, A. I. Gorlin, S. S. Kamynin, D. A. Koryagin and E. Z. Lyubimskiy]

[Text] Principles of constructing an algorithm for planning of calculation chains are described. The algorithm is a modification of the well-known forward wave algorithm that allows substantially reducing the number of nodes of the exhaustive search tree. Bibl. of 5 titles, 1 figure.

UDC 51 : 681.3.06

TEXT EDITOR WITH DEVELOPED SYSTEM OF COMMANDS FOR YES OS

[Abstract of article by N. N. Bezrukov, Kiev]

[Text] Described in the article is the NEATED editor that has a powerful and flexible system of commands. Realization of the system of commands is based on using symmetrical lists. Structurally, the editor is constructed as an interpreter of a one-pass programming language: the syntactic analyzer controls the block of lexical analysis and semantic subroutines. Bibl. of 16 titles.

UDC 681.3.014.001.63

TOOLS FOR MODELING REAL-TIME SYSTEMS FOR THE 'EL'BRUS' MULTIPROCESSOR COMPUTER COMPLEX

[Abstract of article by K. G. Suleymanov, Gomel']

[Text] Described in the article is a set of program tools for designing real-time programs for the "El'brus" multiprocessor computer complex. Programs are designed in the specially developed OPAL language. The tools have been realized on the BESM-6 computer and consist of a translator from OPAL and a model of the "El'brus" multiprocessor computer complex which includes a model of the OS. Bibl. of 16 titles, 1 figure.

UDC 681.3.06

DETERMINING BLOCKING FACTORS OF JOINTLY PROCESSED FILES

[Abstract of article by Yu. V. Trifonov, Gor'kiy]

[Text] Described are the statement and methods of solution of the problem of selecting blocking factors of files located in direct access storage and used in several jobs. Bibl. of 10 titles.

UDC 681.3.06

REALIZATION OF THE PASCAL LANGUAGE FOR THE 'EL'BRUS' MULTIPROCESSOR COMPUTER COMPLEX

[Abstract of article by S. V. Vdovkin, A. A. Kubenskiy, S. S. Lavrov and V. O. Safonov, Leningrad]

[Text] Described are the source language of the PASCAL translator for the "El'brus" multiprocessor computer complex and the general principles of realization and structure of the translator. Some characteristics of the translator and an evaluation of realization are given. Bibl. of 5 titles.

UDC 681.3.06

PROGRAM REALIZATION OF INFORMATION RETRIEVAL SYSTEMS ON SYSTEM OF SMALL MINICOMPUTERS

[Abstract of article by A. I. Kitov, V. A. Litvinova, Ye. I. Dubinina and V. N. Taralova]

[Text] Described are the structure, algorithms and modes of operation of the NORMIN IPS [information retrieval system] realized by using the MAMPS (DIAMS) operating system. DIAMS has been adopted on Soviet SM type minicomputers and has a number of advantages with respect to construction of information systems.

The NORMIN IPS uses normalized natural Russian language for representation of texts of documents and queries. Bibl. of 4 titles, 7 figures.

UDC 681.142

SIMULATION OF OPERATING SYSTEMS

[Abstract of article by N. A. Krinitskiy and T. F. Chernova]

[Text] A model of an operating system and the language for describing the simulated object are discussed in this article. Bibl. of 13 titles, 1 figure.

UDC 681.142

CHARACTERISTICS OF THE IJK TRANSLATING COMPLEX

[Abstract of article by O. I. Rau]

[Text] Discussed in the article is the ideology of a complex that is intended for translation of programs from various source languages for various machines in a computer network and in various programming systems. Bibl. of 11 titles, 1 figure.

FUNCTIONAL ORGANIZATION OF THE ATTENTION ROUTINE AND MANAGEMENT OF PRIORITIZED SELECTION OF PARTITIONS IN THE YES DOS OPERATING SYSTEM SUPERVISOR

[Abstract of article by Yu. D. Kolyakin and S. A. Vaganov, Magnitogorsk]

[Text] Described is the functional organization of the attention routine of the YeS DOS supervisor that expands the set of operator instructions, and the instructions for management of prioritized selection of partitions. Bibl. of 1 title, 2 figures.

8545
CSO: 1863/226

UDC 681.3.06

MICROPROCESSOR SYSTEM DEVELOPER'S TERMINAL

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 7, Jul 81 pp 4-6

[Article by A.V. Bogachev, engineer, and S.S. Zabara and A.D. Mil'ner, candidates of technical sciences]

[Text] Microprocessor facilities represent a new and promising trend in the development of data processing and control facilities [1]. But the application of microprocessor facilities simplifies systems based on them, but not the design process itself. The designing of microprocessor systems (MPS's) represents the development of hardware based on a selected microprocessor and the creation of software--PO (system and applied programs) which implements specific functions of the system. Furthermore, programming and the debugging of programs comprise the major and most labor intensive portion of the design process. The proportion of costs for the development of software is growing steadily and, according to estimates of the majority of specialists, has reached 90 percent of the cost of the entire system [2].

The extensive introduction of microprocessor facilities for controlling different types of equipment and systems in real time and the need to reduce costs for the development of software for them necessitate the creation of special automated design facilities, so-called microprocessor system design systems (SPM's).

One of these systems is that developed at the Scientific Research Institute of the Design of Microprocessor Design Systems (Kiev) and implemented on the basis of a problem-oriented complex (POK) for programming microprocessor equipment--the POK ASPROM [automated system for designing microcomputers], executed on the basis of an SM-4 minicomputer with its real-time operating system (OS RV). The design of a specific SPM based on the POK is accomplished by the addition of special packages of programs and certain special hardware modules. Functional hardware and software of the basic POK ASPROM are primarily adjusted by means of these facilities for operation with a specific MPS. The use of the POK ASPROM makes possible two main steps in the debugging of software and microsoftware (software, below): isolated and combined.

Isolated debugging is performed by the software simulation (interpretation) of programs at the level of the machine representation of a specially created language, MIKROSLENG (practically at the source level). Here the checking of algorithms and computing processes is carried out with the revelation of the main mass of relatively simple errors. But the software interpretation and simulation of the functions of a microprocessor cannot provide a guaranteed check of programs [3]. Combined debugging and a check in the actual MPS being designed are required. For this purpose it is necessary to enable interaction between the SPM and MPS being designed for the purpose of copying programs debugged in isolation, entering into the MPS data simulating the functioning of external objects and for reading out to the developer (operator) the results of checking programs on real MPS equipment.

The fulfillment of these requirements is made possible in the POK ASPROM by the presence in its structure of a developer's terminal (TR), which is discussed in this article.

Design Concepts and Capabilities

The following are the key concepts in the design of the developer's terminal: dispersed processing of data in the SPM, dialog interaction between the developer and the SPM and the MPS being designed, and universality.

By universality is meant the ability to tune the developer's terminal to a microprocessor of a specific type from a broad class of microprocessor facilities just by connecting an additional emulation module.

For many MPS's a check on real equipment is performed by the intrasystem emulation (VSE) method [4]. This affords the developer the ability of access and control required for working with a microprocessor and its internal registers, as well as an emulation storage for the purpose of modifying programs in the debugging process:

By the VSE method, which assumes emulation (substitution) of the microprocessor of the MPS being debugged by its hardware analog (instrument microprocessor), built into the emulation module.

By direct control, i.e., by direct coupling of the developer's terminal with different points of the equipment of the MPS being debugged.

The instrument microprocessor is connected to the MPS by means of a cable terminating in the pin section of a connector built into a jack in the MPS microprocessor being emulated. Direct control is accomplished via a coupling module connected by means of prods to various contacts of the integrated microcircuits of the MPS being debugged.

Emulation of the storage of the MPS being debugged means the substitution of the entire storage or a certain area of it with an individual instrument storage contained in the emulation module and not dependent on the storage of the SPM. The instrument storage is furnished with means of connecting it to the internal line of the MPS being debugged with the presence simultaneously of the ability of access to the latter through the internal line of the developer's terminal, which makes possible the modification of programs in the debugging process.

The terminal possesses the following capabilities: of debugging and simultaneously testing programs and equipment; of step-by-step tracing of the execution of a program; of tracing the implementation of programs in the program tracing mode; and of executing programs in real time.

In the program tracing mode the developer's terminal runs the program in real time, tracing addressing limitations and the identification of registers specified by the developer in debugging. Here the tracing facilities (trace storage) make possible the storage of 128 operations executed by them.

The terminal includes a logical analyzer for analyzing address lines and data lines (up to 32). When the developer assigns a halt point the state of lines is reproduced in 64 cycles preceding the specified halt condition. This makes it possible to establish through which real program the processor reaches a specific state and where control is them transferred to.

The terminal makes it possible for the developer, after isolated debugging of programs on the central computer, to load (copy) a prepared program into the instrument storage of the developer's terminal and to carry out combined debugging in the dialog mode. The program checked can then be transferred to the central computer for putting out technological documentation (wiring tables, punched tapes for a mask, etc.) or upon the developer's instruction the developer's terminal "programs" (burns in) the integrated circuits of the ROM.

In addition, the presence in the structure of the developer's terminal of a general-purpose processor, a working storage of sufficient capacity and a display makes it possible to use it for the preparation of texts and for editing programs in the independent mode.

Structure and Composition

The developer's terminal is in the form of a modular design whose basis is an I41 internal interface (multibase). The central section of the developer's terminal is constructed on the basis of an SM-1800 microcomputer whose universal processor (K580) makes it possible to control all modules and peripherals of the developer's terminal. A second processor is included in the emulation module and performs the functions of the processor of the MPS being developed.

A block diagram of the developer's terminal is shown in fig 1. The module for coupling with the OSh ("common line") accomplishes coupling between the developer's terminal and SM series computers which have an output to an OSh interface. Exchange is accomplished through direct access to the storage with words of 16 bits each with a speed of up to 200 K bytes per second over a distance of up to 10 m.

The floppy disk (NGMD) external storage is implemented on the basis of a dual R1X45D storage unit and is designed for storing the operating system. Source program modules or object modules produced in the debugging process can also be stored in the floppy disk storage unit in library form.

The I/O modules make it possible to connect to the developer's terminal peripherals, including a tape punch station, a printer (PU) and a display module. All control and debugging instructions are given by the operator (developer) from a keyboard and all information regarding the debugging process is displayed on the screen of the display module.

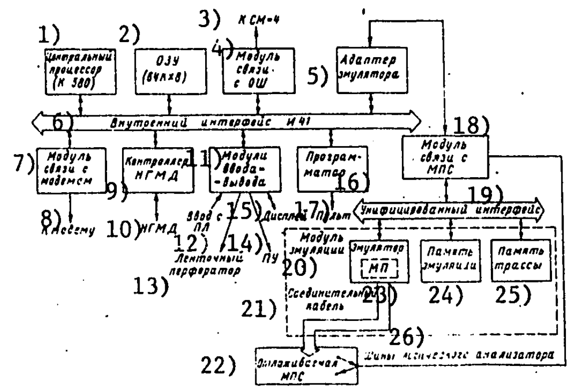


Figure 1.

Key:

1. Central processor (K580)
2. RAM (64 K X 8)
3. To SM-4
4. Module for coupling with OSh
5. Emulator adapter
6. I41 internal interface
7. Module for coupling with modem
8. To modem
9. Floppy disk storage controller
10. Floppy disk storage
11. I/O modules
12. Input from punched tape
13. Tape punch
14. Printer
15. Display
16. Programmer
17. Console
18. Module for coupling with MPS
19. Unified interface
20. Emulation module
21. Connecting cable
22. MPS to be debugged
23. Emulator, MP [microprocessor]
24. Emulation storage
25. Trace storage
26. Logical analyzer lines

The programmer console makes it possible to enter the codes of the debugged program into the ROM, implemented with series K556RT4 microcircuits (K556RT5's are proposed in the future).

The module for coupling with a modem makes it possible to couple the developer's terminal with synchronous and asynchronous modems or signal conversion equipment having an output to an S2 interface. The presence of this module in the composition of the developer's terminal makes it possible to place the terminal at a considerable distance from the central computer.

The developer's terminal also includes an intrasystem emulator (VSE) which operates in real time and makes possible direct coupling between the MPS to be debugged and the SPM. The intrasystem emulator is functionally and structurally divided into two modules: one for coupling with the MPS and one for emulation. One represents the hardware of the VSE, independent of the type of microprocessor (MP) in the MPS, and the other includes the instrument microprocessor and the emulation and trace storage units and is replaced when changing the microprocessor in the MPS. This module is conneced through a unified interface.

The terminal's software consists of a software system (SPO) and test facilities. The former includes a monitor routine, linkage routines (module drivers), a text editor program, a debugging routine, and routines for servicing floppy disk libraries. The test facilities consist of a basic combination of tests and a set of tests. The basic combination is designed for testing the key instructions of the central processor (TsP), loading instructions and instructions for starting the TsP's test, and individual tests for servicing system equipment. The set of tests serves the purpose of checking the working order of modules of the developer's terminal and of locating malfunctions.

Key Technical Characteristics of Developer's Terminal

Central Processor	
Word length in bits	8
Clock frequency in MHz	2
Maximum capacity of storage in bytes	64 K
Instruction set	Determined by architecture of K580IK80 microprocessor
Display	
Number of:	
Characters in a line	80
Lines	24
Character set	96
Floppy Disk Storage	
Capacity of storage in bytes	2 X 256 K
Average access time in ms	483
Common Line Interface Module	
Data rate in bytes/s	200 K
Distance from computer in meters	10
Setting of check points	Up to 2 (for address; one of two addresses; data; address and data; state of testers; chassis ground)

Connection of testers:	
Input	16
Output	16
Capacity of emulation storage in bytes	16 K
ROM programming	K556RT4
Supply voltage in volts	220 + 10%, - 15% (50 ± 1 Hz)

The keyboard is alphanumeric, digital and functional. The debugging mode is step-by-step, cyclic, tracing, real time. The structural design is in the form of a table and pedestal.

The key design concepts and the capabilities of the developer's terminal have been verified at the present time. Development is under way on a number of "user's" developer's terminals tuned to the most popular microprocessor sets in practice, series K580 and K589.

Bibliography

1. Baums, A.K., Gurtovtsev, A.L. and Zaznova, N.Ye. "Mikroprotsessornyye sredstva" [Microprocessor Facilities], Riga, Zinatne, 1979.

2. Uotson, R. "Comparison of Industry-Produced Programming Facilities for Microprocessors," TIIEIR, Vol 64, No 6, 1976.

3. Lipayev, V.V. and Kaganov, F.A. "System for Automating the Technology for the Development of a Set of Control Programs for Microprocessors and Microcomputers," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 1, 1980.

4. "Diverse Design Facilities," ELEKTRONIKA, No 8, 1976, pp 57-64.

8831
CSO: 1863/216

BRIEFS

RELIABILITY OF ON-BOARD SYSTEMS--This mathematical article describes the use of statistical methods to determine the accuracy of on-board automated control systems. These methods are classified in terms of the task, the known information and the equipment utilized. [By B. S. Sinitsyn, doctor of technical sciences] [Editorial Report] [Moscow IZMERENIYA, KONTROL', AVTOMATIZATSIYA in Russian No 3, Mar 81 pp 43-52]

CSO: 1863/252-A-P

APPLICATIONS

UDC 002.6

PROVIDING INFORMATION FOR SCIENTIFIC RESEARCH ON COMPLEX PROBLEM OF 'POWER ENGINEERING' BY DATA BASE TELEACCESS

Moscow VESTNIK AKADEMII NAUK SSSR in Russian No 6, Jun 81 pp 12-17

[Article from the section "Reports of the Presidium of the USSR Academy of Sciences"]

[Text] A terminal has been installed at the Institute of Atomic Energy imeni I. V. Kurchatov that allows institute associates teleaccess to the information file of the International Center of Scientific and Technical Information (MTsNTI) [ICSTI] to retrieve information from this file on the complex problem of "Power Engineering in the interactive mode. The system has been implemented on standard hardware and uses the standard programs for the YeS EVM [Unified System of Computers]. At a meeting of the Presidium of the USSR Academy of Sciences, L. N. Sumarokov, director of the ICSTI, reported on the service provided to IAE associates in this teleaccess mode.

L. N. Sumarokov noted that modern information technology provides for making up a common computer information file--the so-called data base--from the various domestic and foreign source files, automated on-line output of information items on specific problems and teleaccess to the data base, including retrieval in the interactive mode that allows successive refinement of the query as answers are received. Such retrieval is performed in the section of the information network linking the ICSTI and the IAE.

The information base in the ICSTI system includes a file of secondary documents prepared by the VINITI [All-Union Institute of Scientific and Technical Information = AUISTI], a file of R&D reports and a number of international files. The system has been implemented on the YeS-1040 computer. The software includes the standard operating system 6.0, the "Kama" operating system for management of the data teleprocessing mode and the "Dialog" applications program package.

The ICSTI computer file now holds about 500,000 documents on the subject of "Power Engineering" with an annual increase of about 300,000 documents. Thanks to the use of modern information technology, a group of 8 to 10 people is sufficient for file management and operation.

The AP-64 terminal, included in the YeS computer peripheral set, was installed in the IAE.

The file is accessed through telephone lines. Retrieval in the interactive mode from the individual subfiles with a size on the order of 100,000 to 150,000 documents (all information is divided into such subfiles due to the limited amount of disk storage) takes 15 to 20 minutes. About 3,000 retrieval sessions per year can be performed using the two terminals now in operation. The experience gained at the ICSTI while servicing IAE associates is of interest in many respects.

The fact is that in the next two to three years, information organizations may make up and operate retrieval files containing about .5 to 1 million documents, that is files comparable to the ICSTI "Power Engineering" file (disk storage size is the limit). At this stage it is probably advisable to begin developing problem-oriented files on the major areas of science and technology defined by the comprehensive development programs: power engineering, fuel, raw materials, agriculture, biology, food, light industry, transportation, etc.

The necessity of continuing to assimilate existing information technology and means of accessing information and expand the scale of remote retrieval in the interactive mode, i.e. real access to files, is becoming obvious.

ICSTI is now organizing service for the Institute of Radioelectronics (the terminal here will not be a subscriber station, but a minicomputer that substantially expands the capability of retrieval and processing of data in the teleaccess mode), the Institute of Bioorganic Chemistry imeni M. M. Shemyakin, the Moscow Engineering Physics Institute and other institutions.

Another serious problem is occurring in connection with what has been said. In addition to the files of scientific and technical information being created in the information agencies, services and subdivisions, the leading institutes are forming so-called factual data banks based on the results of their own research and development. Unfortunately, these banks created at the institutes are not registered anywhere today and the very formation of them is not coordinated with anyone. As a result, access to these banks is made difficult for "external" organizations. Solving this problem would raise sharply the efficiency of making use of scientific research results.

Also, it is not simply a question of the division of the range of problems, but of a unified, coordinated data compilation plan that guarantees each organization's bank will be accessible to all others. In addition, acquiring and processing integrated information and building unified banks of scientific, technical, commercial and market information are a separate problem.

After the report was made, A. P. Aleksandrov, acamedician and president of the USSR Academy of Sciences, remarked that IAE scientists were still not used to using the terminal, but that those who had mastered it were very satisfied. It is used especially often in preparing dissertation works and reviews.

Continuing this thought, L. N. Sumarokov said that usually work on each new subject begins with a thorough retrospective retrieval of information on a certain range of problems. As for habit, the case is the same as in general when a computer is introduced in a scientific institution: Those who have occasion to use its services at least once usually continue doing so later on and in time the number of users builds up. After all, modern display terminals are rather simple and require no special knowledge of programming and computers. Questions are formulated essentially in natural language and refined over the course of the retrieval; upon request, you can quickly get a printout of any documents of interest.

After his report, L. N. Sumarokov was asked a number of questions.

Replying to the question by the late academician N. V. Mel'nikov on the possibility of comprehensive analysis of information, forecasting and preparation of analytic surveys, L. N. Sumarokov said here it is only a question of cooperation between information specialists and analysts. Information specialists can only prepare very enriched resources for analytic scholars. Analytic surveys, prepared on the basis of information from ICSTI files, were demonstrated in an exhibit at the Presidium of the USSR Academy of Sciences.

Academician P. L. Kapitsa stressed that using teleaccess to files of information in machine-readable form makes it possible to not only compensate for a reduction in the inflow of literature, but also to achieve more expeditious distribution of information than the traditional periodicals.

L. N. Sumarokov responded to P. L. Kapitsa's request for a teleaccess demonstration on the premises of the Presidium that ICSTI specialists had demonstrated that before and it was considered inadvisable to repeat it at this meeting of the Presidium.

In answer to academician A. V. Fokin's question, L. N. Sumarokov reported that essentially any text appearing on the display screen could be printed out upon the user's request. Moreover, the user may obtain not only the text of the abstract shown on the display, but also a microfiche or full-size copy of the original report itself--if, of course, it is available in the institute's holdings. The convenience of the interactive system is that thanks to gradual refinement of the query as information comes in, the user may obtain a printout of just those documents that are of immediate interest to him.

The information files are compiled from the most varied sources, the speaker noted in answering academician M. S. Gilyarov's question. The data files containing factual information are built from the data banks of various leading institutes and several commercial data banks purchased by the center. The basis of the scientific and technical information files containing reports on scientific publications is formed by the AUISTI's REFERATIVNYY ZHURNAL (RZh) in machine-readable form, a domestic file of information on scientific research work performed and several other files. In all, the ICSTI information base now combines about 10 source files, including foreign files, each of which is continually being supplemented. The technological problem in building a unified data base on the basis of these different files consists primarily in achieving compatibility of the various files, recorded in various formats and organized in various ways. The capability of accessing the complex of these heterogeneous files as some unified information file has to be provided.

Concerning academician P. L. Kapitsa's comment on the need for verifying the validity of the information input to the system, L. N. Sumarokov expressed the opinion that such checking is rather the job of scientists, and not only information specialists.

A. I. Mikhaylov, director of the AUISTI, took part in the discussion on the report; he noted the progressiveness of information service using a terminal system. The AUISTI contributes to the organization of this service by sending information agencies large information files in machine-readable form. It now puts out on magnetic tapes abstract journals on biology and chemistry, and physics is in the preparation stage. On the whole, about 700,000 secondary documents are now put out on magnetic tapes. A priority task is organization of output on magnetic tapes of information publications on machine building. (Academician A. P. Aleksandrov, president of the USSR Academy of Sciences, remarked during the statement that files on materials technology and metallurgy, especially on strength and resistance to corrosion, are absolutely necessary.) With sufficient capacity to transfer the entire AUISTI REFERATIVNYY ZHURNAL to magnetic tape and with sufficient amount of machine storage, information could be grouped by any combination of features, forming magnetic tapes on any direction of research, any problem of interest to consumers. In general, in A. I. Mikhaylov's opinion, the main path in development of scientific information activity is the establishment of in-house data bases on magnetic tapes from primary sources available in domestic holdings.

Unfortunately, the terminals available at the AUISTI are capable of operating only st short ranges and the image quality on the screens is very poor. AUISTI has developed an automated system that is to operate in the mode of a network and terminal links. Over the five-year plan, 22 information centers are to be connected to this system. However, the lack of necessary equipment and, mainly, communication lines is hindering the realization of this system. But the fundamental and technical problems of teleaccess have essentially been solved: The "NTI-80" all-union exhibit was in operation during the fall at the USSR VDNKh [Exhibition of Achievements of the National Economy]; a terminal was installed there that allowed entry to the data banks of the AUISTI, the INION [Institute of Scientific Information on the Social Sciences (USSR Academy of Sciences)] and the sector center for instrument building and automation equipment. Special communication lines were allocated for this and the system operated quite satisfactorily.

In reply to A. P. Aleksandrov's comment on the possibilities of using the SM-4 minicomputer as a terminal, A. I. Mikhaylov said that this machine has only 64 characters and it is very difficult to send formulas using it. Hungarian video terminals are preferred at the AUISTI.

To improve the information service for academy specialists, the technical base for scientific information activity now has to be developed first of all.

Ye. P. Velikhov, vice president of the USSR Academy of Sciences, stressed the importance of the work described in the report as one of the first examples of industrial operation of a system linking the information consumer in a scientific institution to the data banks in an information agency. This is found to be very efficient even with existing equipment with all its shortcomings. And in this case, Ye. P. Velikhov stressed, the institutes have to be encouraged to connect to these systems. The simplest way to encourage this is to establish a centralized resource for the data equipment.

Academician N. G. Basov supported the proposal to set up a centralized fund and added that the effort to build the data banks has to be coordinated and planned, in particular to build banks of various physical constants and measurements, for example, for optics and spectroscopy. N. G. Basov stressed that we generate a lot of data in this field, but nobody is collecting, systematizing and recording it.

Another aspect of "information technology" that the academy should get involved in is the recording on machine media of operating manufacturing instructions. Abroad in general, working drawings are now often a thing of the past. The data needed is recorded on perforated tape which is then used to control the operation of automatic machine tools. When necessary, the information recorded on tape may be transmitted over communication channels, even between continents.

Also very promising is the use of small sections of magnetic tape for recording commercial, bookkeeping, accounting, etc. documents. For example, in England, a considerable portion of office paperwork is now duplicated on magnetic tape. This practice greatly simplifies automation of accounting, acquisition of statistics, office records, etc., while the recording on magnetic tape of invoices can simplify many warehouse operations.

A. P. Aleksandrov remarked that such efforts in the academy are underway jointly with a number of ministries. Test sites are being set up and groups of automated machine tools with remote data entry and centralized control installed in them. And the first automated warehouses are operating in the country.

Academician N. P. Fedorenko supported the suggestion of encouraging formation of an information network within the framework of the USSR Academy of Sciences. One way of forming a unified information network for the Academy of Sciences of the USSR is to establish collective-use computer centers to serve a number of institutes. Institutes today are trying to set up their own computer centers which is an inefficient waste of equipment. A strategy for utilizing computers within the entire academy has to be worked out. In particular, the Section of Social Sciences and the Central Economic Mathematics Institute of the USSR Academy of Sciences (TsEMI) at one time proposed setting up a collective-use computer system for the scientific institutions of the social sciences on the base of the TsEMI VTs [computer center].[1]

G. S. Pospelov, associate member of the USSR Academy of Sciences [AS], remarked that the problems of the strategy for using computers to handle scientific information tasks will soon be discussed in the Coordinating Committee on Computer Technology. However, formation of a unified information computing network for the AS must be preceded by a large effort to establish the overall concept of distribution of academic data banks and to structure the files of scientific information.

Academician Yu. A. Ovchinnikov, vice president of the USSR AS, noted the importance of the work carried out in the ICSTI and IAE. The results of this work are helping introduce a new concept of information work in the scientific institutions and train scientific workers in using modern methods of information service.

Yu. A. Ovchinnikov emphasized the impossibility of establishing data banks resting exclusively on domestic files in the institutes, for example, of a biological or chemical type. Where possible, foreign files must also be used along with domestic

---

[1] See: VESTNIK AN SSSR, 1977, No 9, pp 74-86.--Editor

data bases, for example, the abstract file of the AUISTI on magnetic tapes. Thus, not one of the accessible domestic and foreign files corresponds to the problems of the Institute of Bioorganic Chemistry imeni M. M. Shemyakin--a different classification is needed, a different approach. The institute expends a great deal of effort on classifying and entering information into its own bank. But in return, when the files coming into the institute are reprocessed with the participation of its associates, a unique information file is obtained in the output. In principle, it is quite possible to sell this file abroad--the value of it is very great.

It is also very important, Yu. A. Ovchinnikov stressed, that all the work that L. N. Sumarokov reported on is being performed on domestic equipment. The experience of using this equipment for solving scientific information problems and for working out interactive retrieval in the teleaccess mode is extremely useful for developing this form of information service in other USSR AS institutions.

In recent years, Yu. A. Ovchinnikov continued, scientists have begun to better understand how necessary it is today to get the scientific information they need. As before, we are not the first ones who do not like to leaf through journals. As before, we all like original articles and we always enjoy the chance of working with them. But it is clear that this is not the main route to obtaining information. And the dissemination of the know-how gained at the ICSTI is an urgent problem; it is useless to expect some special decrees or meetings to solve it. The bureau of departments has to take decisive measures for this.

At the end of the discussion, academician A. P. Aleksandrov, president of the USSR AS, emphasized the need of a unified policy in developing information and computer systems in the USSR AS that achieves both economy of resources as a result of introduction of standard developments and compatibility of the information systems being created.

In the resolution adopted, the USSR AS Presidium noted that the activation of the "Institute of Atomic Energy imeni I. V. Kurchatov--ICSTI" section of the information network raises information support to research on the complex problem of "Power Engineering" to a qualitatively new level and approved the work performed by the ICSTI on developing the information network for teleaccess to data bases of scientific and technical information.

The presidium considered it advisable to disseminate the expertise of this work and instructed that a proposal be prepared to form a centralized fund to finance it in the institutes of the USSR AS.

8545
CSO: 1863/215

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

UDC 622.692

USING MICROCOMPUTERS IN PUMP STATION CONTROL SYSTEM FOR MAIN OIL PIPELINES

Moscow AVTOMATIZATSIYA I TELEMEKHANIZATSIYA NEFTYANOY PROMYSHLENNOSTI in Russian No 6, Jun 81 (manuscript received 28 Apr 81) pp 10-11

[Article by K. S. Bogdanov and B. M. Prokhorov, VNIIKAneftegaz [All-Union Scientific Research, Planning and Design Institute of Complex Automation in the Petroleum and Gas Industry]]

[Text] Developing automated process control systems to pump oil through main pipelines involves using microcomputers in the pump station control system (SUNS). Domestic industry is assimilating production of microcomputers such as the "Elektronika S5-01," the single-board "Elektronika S5-11" and "Elektronika S5-21," the PS-300, the "Elektronika NTs-0.3" and the "Elektronika-60."

The pump station control system, as a rule, contains the following basic subsystems:

the subsystem of automatics that effects program control of starting and stopping the station pump units, program control of starting and stopping auxiliary mechanisms, automatic protection of the pump units and station as a whole and the equipment status alram system;

the subsystem for maintaining the station pressure parameters;

the subsystem for metering and monitoring electricity and the complex of electrical protection for the equipment in the electric substation; and

the subsystem for transmitting information to the regional dispatch point (RDP). As a rule, this device is the control point for the system of telemechanics (KPTM) for the main oil pipeline.

In each subsystem, a microcomputer performs its functions.

In the telemechanics system control point (KPTM), the microcomputer executes the following programs:

distribution of time between the various problems in accordance with specified priorities;

formation of buffer files and initial processing of data;

control of operation of devices;

realization data compression algorithms;

formation of messages for transmission over communication channel;

control of reception-transmission as a function of the structure of the communication channel; and

data conversion for reproduction of information.

In the electricity metering subsystem, the microcomputer has to perform:

computation of summary half-hour active and reactive power in the hours of operation of the power system with maximum load;

storage in memory of the maximum values of summary active and reactive power in the hours of operation of the power system with maximum load; and

computation of power consumption as a whole.

As is known, a subsystem for program control of a pump station may be made by various methods: using only hardware for automation without using a computer; using a computer in which the basic functions and tasks of the subsystem are realized by software; and using both a computer and automation hardware jointly.

The authors compared and chose the best structure for building a pump station control subsystem by the minimum of resulting outlays for system operation based on the known technique for selecting hardware complex structure.*

The analysis and comparison of three considered alternatives for building a pump station control subsystem showed that the most expedient is the version without computers (i.e. without microcomputers).

This is also confirmed by the following considerations:

1) computer operations are not required to effect protection of technological equipment, the alarm system and input of reserve; and

2) each channel of a logic device has a limited number of sufficiently reliable elements. Calculations show that the channel-by-channel reliability of the logic automat is higher than the reliability of these channels when a microcomputer is used.

Specialized logic devices have individual control channels while a microcomputer effects all actions through a central processor.

Using a microcomputer in the subsystem for maintaining pressure parameters in a pump station is advisable since analog regulation systems under the conditions of operation in pump stations have a number of fundamental shortcomings. The specific nature of operation of a system that regulates delivery of a pump station is that it operates in the mode of limiting pressure from the top and therefore operating pressure is always below the control setting.

---

* Bogdanov, K. S. and Prokhorov, B. M., "Technique for Selecting Hardware Complex Structure for Oil Pipeline Control System," SNT [Collection of Scientific Transactions], "Problemy razrabotku sredstv avtomatizatsii i avtomatizirovannykh sistem upravleniya v neftyanoy i gazovoy promyshlennosti" [Problems of Development of Automation Equipment and Automated Control Systems in the Petroleum and Gas Industry], Institute of Automation of the USSR Ministry of Instrument Making, Automation Equipment and Control Systems, Kiev, 1977.

Building a microcomputer-based subsystem will allow qualitative improvement of its functioning.

From a consideration of the capabilities of using microcomputers in a pump station control system, it follows that it is advisable to make use of microcomputers in those subsystems where performance of complicated logic operations of the computing process is required with storage of its results and generation of control actions. Using microcomputers just for elementary problems (like starting and stopping pump units or auxiliary mechanisms) is unwarranted and economically inexpedient. These tasks can be successfully performed by simple and reliable logic equipment for automation.

The derived results formed the basis of developments of new automation apparatus for main oil pipelines, namely, the UVTK-100 complex of telemechanical devices (TsNIIKA [State All-Union Central Scientific Research Institute of Complex Automation]) and the BLIK-1 complex of pump station automatic equipment (VNIIKAneftegaz).

The control point for the UVTK-100 complex, installed in the pump station, is built with the "Elektronika-60" microcomputer and performs the set of programs mentioned above. The main advantage of the UVTK-100 complex obtained through using the microcomputer compared to the telemechanical TM-120 is the switch to the sporadic principle for transmission of all information. In the UVTK-100 control point, part of the microcomputer memory is intended for technological programs developed in accordance with user requirements.

The BLIK-1 complex of pump station automatic equipment is built on the base of a specialized logic device made with integrated microcircuits and a set of series instrumentation. The main advantages of the BLIK-1 complex compared to the "PUSK-71" apparatus now being produced are as follows:

1) expansion of performable functions through incorporation of units to control inlet chambers and start-up of the scraper and the panel for a local dispatch point;

2) enhanced sensitivity and stability of the pump station pressure regulation system; and

3) a higher level of reliability through incorporation of test monitoring of malfunctions of the logic device and redundancy of its power sources.

In building a pump station control system based on the UVTK-100 and BLIK-1 complexes, the panel for a local dispatch point may be omitted and its functions may be performed by the microcomputer of the UVTK-100 control point with reproduction of information on a display.

8545
CSO: 1863/224

UDC 681.325.5-181.4

ORGANIZATION OF COMPUTER-INSTRUMENT INTERACTION IN SYSTEMS FOR AUTOMATION OF SCIENTIFIC RESEARCH WITH VARIABLE STRUCTURE

Riga AVTOMATIKA I VYCHISLITEL'NAYA TEKHNIKA in Russian No 3, May-Jun 81 (manuscript received 18 Jun 80) pp 81-91

[Article by S. N. Domaratskiy, O. S. Zudin and O. Vaynio]

[Excerpts] A characteristic feature of recent years is the widespread introduction of distributed computer networks into the practice of operation of industrial and scientific organizations. Networks unite centers consisting of several computers that interact with each other and exchange data with the network through a common processor [1]. These centers may be in fixed locations or even on mobile platforms, for example, on ships intended for oceanic research. Radio and space communications may be used to include these centers in the network. Single-center computer networks located on scientific research ships (NIS) of the USSR AN [Academy of Sciences = AS] unite laboratory systems for automation of scientific research (SANI), automatic navigation equipment, meteosynoptic data acquisition systems and data acquisition and processing centers. Laboratory scientific research automation systems contain computers, data input, output and display devices and a substantial number of scientific instruments. A feature of the functioning of the majority of laboratory scientific research automation systems in ship systems is the variable composition of equipment and the need for frequent and rapid reconfiguration of the system in the process of operating with orientation to various experiments. This article is aimed at generalizing the experience of organization of computer-instrument interaction in laboratory scientific research automation systems with a variable structure gained in the process of a joint effort by Soviet and Finnish specialists on developing the Integrated System for Automation of Scientific Research for the USSR AS scientific research ship, "Akademik Mstislav Keldysh," at the shipyard of the "Khollming" AO [joint-stock company] in Rauma (Finland).

In conclusion, let us note that organization of instrument-computer interaction in laboratory scientific research automation systems should be based on standard interfaces; of these, the MEK [International Electrotechnical Commission = IEC] bus is preferred for the majority of ship scientific research automation systems. To match instruments with this bus in the composition of laboratory scientific research automation systems, it is advisable to have several general-purpose IK [interface cards] on the base of microprocessors and BIS [LSI] programmable logic. The suggested structure of the interface cards allows matching to the IEC bus any instruments with parallel or serial input and (or) output of data used in oceanologic research, and in addition, allows expanding instrument functions when necessary. The suggested interface card is used in several laboratories on the "Akademik Mstislav Keldysh" scientific research ship to interface various measuring instruments and sounding devices to computers. The suggested structure of the program that interprets the FSF [functions included between sections S_2 (instrument functions) and S_3 (interface functions) to match data formats] provides protection from an unsanctioned change in the instrument address without reducing the flexibility of the system as a whole.

8545
CSO: 1863/223

UDC 62-658.012.011.565.001.4

COMPUTERIZED VIBRATION TEST CONTROL SYSTEM

Kiev IYERARKHICHESKIYE AVTOMATIZIROVANNYYE SISTEMY OBRABOTKI DANNYKH in Russian 1980 pp 47-52

[Article "Computerized Vibration Test Control System," by Yu. F. Koval', B. Yu. Mandrovskiy-Sokolov, P. M. Siverskiy and A. A. Tunik from book "Hierarchical Automatic Data Processing Systems," Ukrainian SSR Academy of Sciences, Order of Lenin Institute of Cybernetics]

[Text] The automatic vibration test control system (ASUV-3) is based upon utilization of "first generation" control algorithms. The ASUV-3 is designed to generate control signals with a desired spectral matrix for regulation of a three-component vibration process as well as for statistical processing of rapidly varying signals from vibration-acceleration pickups or other high-frequency pickups.

Among other types of systems performing the same tasks, digital vibration test control systems are considered the most promising [1], since with a minimum of adjustment time these systems permit maximum approximation of computed manipulated variables to real processes occurring during transport and under other product operating conditions.

It is necessary to state two assumptions with respect to control of random vibrations: a) steady-state parameter conditions; b) normality of the law of distribution [2].

These assumptions assume that the statistical parameters characterizing the vibration are steady-state parameters and that the vibration state of the object may be considered a superposition of random processes generated by independent sources.

A basic task in the development of vibration-test control systems accordingly consists in selection of the method of producing a random process with the desired spectral matrix at the output of a dynamic system consisting of a number (from 1 to 3) vibrators and a test object.

The ASUV-3 employs the following control algorithms:
adjustment (governs amplitude of control signals for identification mode, adjusts amplifiers;
identification of amplitude-frequency characteristics (for determining matrices of amplitude-frequency characteristics of an object with a three-component force generator);
zero approximation (to compute manipulated variables taking task and amplitude-frequency-characteristic matrices into account);
iteration control (to correct manipulated variables obtained on the basis of preceding algorithm taking nonlinear effects into account) and

analysis (for spectral analysis of signals from vibration-acceleration pickups during testing of the object).

The system is adjusted for test conditions with respect to 16 or 32 matrices of spectral densities uniformly distributed throughout the frequency range.

The following are the basic system technical characteristics:
the ASUV-3 system permits control of random vibration spectra within the frequency ranges 1-250 Hz and 8-2000 Hz;
control is based upon the use of 16 or 32 averaged frequencies;
the dynamic range of control of process amplitude values is 40 dB;
only natural spectral output densities are controlled;
signals are analyzed over a frequency range of 0-10 kHz;
mode adjustment time using the M-6000 computer is approximately 30 min;
control mode assignment error is roughly 10 per cent;
interface capacity is 10 binary characters.

The ASUV-3 system is designed for operation with electrohydraulic and electrodynamic vibration test stands. The system consists of the following series-produced devices: an M-6000 (with 32 kbyte internal memory) or M-7000 (SM-2) control computer with peripherals (Konsul teletype, DZM [expansion unavailable] alpha-numeric printer, SIGD [expansion unavailable]). The system also incorporates the following special-purpose devices: high-speed Fourier transform processor and interface (with vibration test stands). The figure is a block diagram of the ASUV-3.

The purpose of the individual devices is as follows. The computer, the system's central processor, is designed to control all system devices and perform computations in accordance with control algorithms. The computer controls exchanges of data files between the PBPF [expansion unavailable] and communication devices, the BPF processor with respect to job input and other external devices. It should be pointed out that since system mode-adjustment time is governed entirely by computer speed, the use of the M-7000 or SM-2 should decrease this time substantially. System job input and operating mode selection: vibration test stand control - analysis is performed in conversational mode with the system via teletype or Konsul unit.

The SIGD display is designed for visual scanning of initial, intermediate and computer output data. Visual scanning permits monitoring of the computation process during adjustment of the system for the desired mode and the introduction of corrections if necessary. Information is illuminated on the SIGD as graphs in output-frequency coordinates. Graphs may have 16 or 32 ordinates.

The special-purpose BPF processor performs within the system the function of executing the direct or inverse Fourier transforms.

The interface is designed to provide exchange of number files between the computer, vibration test stand amplifiers and vibration-acceleration pickups. The ASUV-3 system has a three-channel communication device. Each channel has a code-voltage convertor, a band filter and an output amplifier with output to the vibration test stand amplifier. The input channel consists of a matching amplifier, a band filter, a scale amplifier, an analog memory and a voltage-code convertor. Convertor capacity, 10 binary digits, is selected taking into account the dynamic range of the vibration test stands (approximately 60 dB). The coefficient of amplification of the normalizing amplifiers is selected on the basis of computation of optimum signal transmission with the transmission coefficient for the entire interface channel near 1.

The frequency range of the transmitted signals is determined by the type of vibration test stand: 1-250 Hz for electrohydraulic and 8-2000 Hz for electrodynamic stands. With the use of a timer the computer controls signal emission frequency. The computer also selects the operating mode of the communication device depending upon the control algorithm.

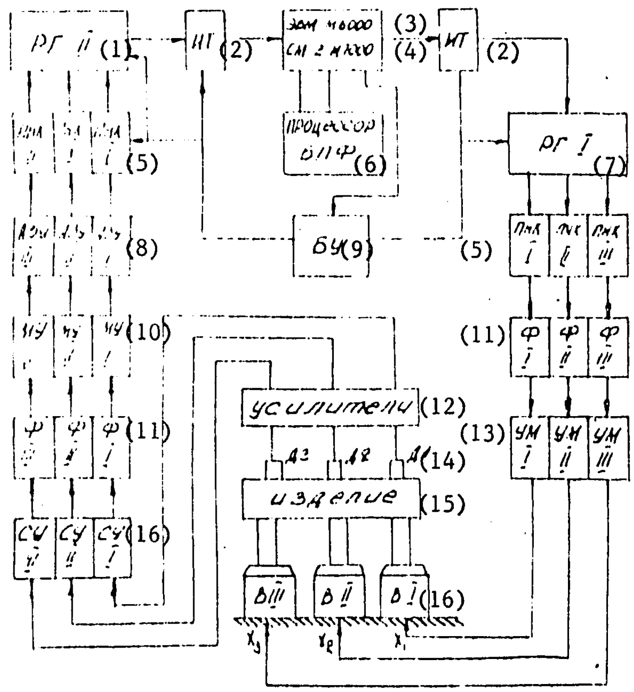


Figure: Block diagram of ASUV-3. 1 - RG [expansion unavailable]; 2 - IT pulse converter; 3 - M-6000 computer; 4 - SM-2 M-7000 computer; 5 - voltage-code converter; 6 - BPF processor [expansion unavailable]; 7 - RG [expansion unavailable]; 8 - AZU [analog memory; 9 - BU [control unit]; 10 - MU [scale amplifier]; 11 - filter; 12 - amplifiers; 13 - UM [output amplifier] 14 - D [meaning unknown]; 15 - object; 16 - V [meaning unknown.

ASUV-3 software is designed to organize automatic vibration-test control and for spectral processing of sensor data. The operating system, which has been developed taking the special characteristics of the operating system of the computer used into account, establishes the sequence in which system operating modes are executed and communication between operator and system. The operating system consists of a control program, programs providing communication between operator and system and special software. Jobs are input for processing both by program through a sequence of control operators and from a control console.

The system and its software are designed so as to permit structural enlargement and the introduction of new control, processing and monitoring modes. The system is thus open for expansion.

FOOTNOTES

1. Getmanov, A. G. et al., "Avtomaticheskoye upravleniye vibratsionnymi ispytaniyami" [Automatic Vibration-Test Control], Moscow: Energiya, 1978.

2. Krendell, S. [Crandall], "Sluchaynyye kolebaniya" [Random Oscillations], Moscow: Mir, 1967.

8963
CSO: 1863/205

UDC 69.003 : 658.3.012.22

USING COMPUTERS IN MAJOR CONSTRUCTION ORGANIZATIONS

Moscow EKONOMIKA STROITEL'STVA in Russian No 3, Mar 81 pp 41-45

[Article by A. F. Sergiyenko, candidate of economic science, chief of the PEU [Economic Planning Administration] of the Glavtyumenneftegazstroy [Main Administration for Construction of Petroleum and Gas Industry Enterprises in Tyumen Oblast]; V. S. Belichev, director of the IVTs [Information-Computer Center] of the Glavtyumenneftegazstroy; and A. A. Mezhlumov, IVTs department chief]

[Text] In 1973, an information-computer center (IVTs) was established in the Main Administration for Construction of Petroleum and Gas Industry Enterprises in Tyumen Oblast; it was based on the YeS 1020 computer with subsequent use of AP-64 interactive terminals; the center was established to improve the organization, planning and management of construction of facilities around the oil and gas deposits.

It was assumed that using computers in the IVTs would first of all help the construction subdivisions of the main administration in the stage of engineering preparation of the construction work. For this, centralized storage in computer memory and use of normative and reference data was planned to be used in forming individual facility and consolidated schedules for construction and installation work, checking the balance of plan tasks with the level of resources on hand and to be allocated, determining the need for materials, equipment and manpower, calculations of normative standard net production and the like. It was assumed that in the IVTs, there would be concentrated all the basic plan, accounting and budgetary calculations and organized the centralized storage of a common reference information base on the course of activity of all subdivisions of the construction organization, the compilation and calculations of network models, the calculations of long-term and annual calendar plans for contract activity, the calculations of the need for resources by the physical amounts of work, the deficiencies by measures being developed and others. It was intended that the IVTs would provide information to all the subdivisions of the construction organization on the numerous questions necessary for normal operation. It was believed that the IVTs activity would make it possible to reduce the number of personnel engaged in the various levels of management of the construction work. Seven years have passed, but the expected effect from the introduction of the computer in construction work implemented by the main administration has not been obtained. In analyzing this situation, many reasons were uncovered that prevented accomplishment of the tasks assigned with the aid of the computer.

First, the construction industry is organizationally and legislatively inadequately prepared for the introduction of complex automated management systems. Thus, it makes no sense to transfer to a computer the existing technique for planning basic tasks and economic indicators only from the level reached, since the existing economic services of the construction organization are successfully handling this with less outlays for labor and resources. And the results of calculations, based on a direct account, using network methods of planning, etc., obtained with the computer when management decisions are made, are purely informative, since they still do not have a developed base. For example, in planning the workers' wage fund it may turn out that substantially fewer resources have been provided for the planned amount of work by budgetary pricing than that obtained in planning from the level reached. However, this still does not mean that the decisions will be made to reduce the workers' wage fund for the planned period, since this may undermine the production economics of the organization. Such a situation may become only the basis for analysis of the reasons the disparity resulted and for finding ways to eliminate them as quickly as possible; such a situation may be caused by both an imperfection in the budgetary pricing system and mistakes in organizing and planning construction work. To introduce into the practice of construction work a planning system based on a direct account, in our view, the conditions required are coverage by this system of planning of all construction organizations in a given region and the presence with that of an approved procedure for calculating plan indicators.

Second, up to now, established plan indicators do not balance with physical resources allocated and existing production capacities of the organizations. For example, if computer calculations show that the work program outlined by the plan cannot be fulfilled with available production capacities, even taking into account the possibility of increasing them, this still does not mean that plan tasks will be reduced when the calculations are presented to the appropriate higher authorities. In such cases, a recommendation is usually made to find reserves of the necessary size to make up for the shortages in production capacities, to raise the level of organization of production, to improve the technology of performing construction and installation work, to introduce achievements of science and new technology, but not to change the plan tasks.

Of course, both in the first and the second case, it cannot be said that the calculations made on the computer were not needed, even though they did not yield the results desired. They were needed to keep management properly informed when decisions are made on managing the construction. The effect of this information may even be considerable, since the director of the organization sees the real picture of the disparity in the decisions made and may forecast his solutions, gradually "pull up" the actual level of organization of production to that stipulated by the computer calculations or find ways to eliminate the deviations noted. However, it must be especially noted that under the current organization of planning and management of construction work, outlays to prepare this type of information are very substantial and at times are not offset by the effect obtained from making use of it.

Therefore, before preparing any information for a computer, there must be thorough consideration of just how necessary it is to making decisions on management and what is the expected economic effect from using it in the practice of managing construction work. In addition, it is not always possible to achieve the required objectivity in the calculations, which lowers the effectiveness of using a computer.

APPROVED FOR RELEASE: 2007/02/09: CIA-RDP82-00850R000400050067-9

Third, the precise calculations made on a computer, for example, for requirements for physical resources are not always the basis for allocation of them, because physical resources are aften allocated being guided by nonformalized considerations or grounds for which manual calculations are quite suitable. Naturally, the question arises, why them are these precise calculations necessary if in other organizations they are trusted in the same way as manual calculations?

Fourth, when the computer was introduced into construction work, neither the organization directors nor middle level managers were prepared for this. None of them got into the potential capabilities of the computer for improving the organization of management of construction; all were occupied with current duties; the needed experience of introducing computers in related organizations was not there; there were gaps in the relations between the ASU developers and the top level of management of construction. As a result of this, the mid-level management staff of the main administration was engaged in using the computer.

The computers were loaded with ank kind of problem to ensure maximum utilization of them. In the first place, there were strong attempts to use the computers to automate office operations: wage accounting, accounting of physical production stocks available in bases and warehouses, compilation of consolidated statistical reporting, etc. In the process, by decisions of the main administration, these functions were not assigned to the information-computer center.

Fifth, the information-computer center was charged with the responsibility of collecting the information for all subdivisions of the main administration, but was given no rights at all to ensure this acquisition. As a result, the main administration subdivisions did not deliver the information to the IVTs on time and the quality was poor.

Sixth, implementation of the tasks described above was to have achieved a reduction in labor outlays and numbers of administrative and managerial personnel in the subdivisions and staff of the main administration. But this did not happen; the number of workers is not being reduced, but growing, in particular, through staffing of the information-computer center. The labor outlays for the noncentralized creation of the program and information base for the tasks mentioned above through the efforts of just the information-computer center are very substantial, and the introduction and use of them in the construction industry unprepared for automation are so ineffective that this must be stated: the computer capacities in the main administration at this stage of development of production relations established in construction are great and must be reduced.

It seems to us that the same situation with the use of large computers exists in other major construction organizations too and this question should not be ignored. It is not a question of whether or not it is necessary to ease the work of economists or accountants, since hardware facilitating accounting and computing operations is badly needed here. But for these operations there is small hardware: microcomputers, minicomputers, accounting machines, data recorders, etc.; for now it is simply inefficient to use for these purposes large computers with the modern level of development of peripherals and networks of communication in construction organizations.

As is known, construction is a very complex economic mechanism because of the variety of processes and resources fitted in them, the plurality of participants in the process and the diversity of their activity, and the peculiarities of the conditions for performing work, in particular, in Western Siberia. These complexities engender a host of unresolved problems in equipment, technology and organization of construction. To solve many of these problems, computers, of course, are not needed, since they are most often solved by simply following the rules of the SNiP [construction standards and regulations], and observing the decisions of the work organization plan and work accomplishment plan (POR and PPR). But there are a number of problems where a computer would be invaluable, for example, for centralized storage of common reference information on the course of activity of all subdivisions of a construction organization, compilation and calculation of network models, the labor plan, long-term, annual and calendar plans for contract activity, need of resources by physical amounts of work, miscalculation of measures on developing oil and gas industrial deposits, etc.

Information of a varying degree of consolidation is needed to make a number of plan calculations based on the normative base that exists in construction and for the level of the main administration. The computer can be an invaluable aid in consolidating information too. Here roughly are those questions in which, in our opinion, it is necessary to use a computer in construction organizations.

Computer operation practice in the main administration shows that on the scale of a construction main administration, it is necessary to establish an IVTs with a powerful hardware base for processing, storing, transmitting and displaying information, a ramified network of communication throughout the entire region and centrally developed machine programs that reflect approved techniques for calculating technical and economic indicators. The activity of the IVTs is currently being restructured at the main administration to raise the efficiency of using computers. An effort is underway to set up a centralized bank of normative and reference information as the base for building a management information system for the main administration for decision-making. The hardware for the information-reference system is based on two YeS 1033 computers and data teleprocessing equipment, in particular, by using the AP-64 and AP-70 interactive terminals. Using these interactive terminals equipped with displays, the main administration management staff can have access to all computing facilities and become a participant in the computing process. As a rule, workers at the lower and mid levels of the management staff of the main administration compile the appropriate information for preparing managing decisions. It is prepared by various functional subdivisions of the main administration and by each office individually through its own channels and methods of acquisition; therefore, the same information coming in from various sources is often contradictory and unobjective. Creating the centralized normative-reference bank using the computer hardware complex and data teleprocessing equipment will make it possible to put preparation of information in good order and preclude cases of contradiction, distortion and duplication of information.

Workers in the lower and mid levels of the management staff of the main administration will have to continually maintain the centralized information-reference bank in working condition by independently adding to and correcting the information stored in the computer. As a result, all workers on the management staff of the main administration will be using common information.

At the same time, these very workers themselves become participants in the automation process. While expending their labor to create the information base, they encounter the computer hardware and the specific problems and difficulties in collecting and processing the information which are solved more simply by these same workers on the management staff.

In addition, the IVTs is performing the preparatory work associated with carrying out the decisions stipulated by the decree of 12 July 1979 by the CPSU Central Committee and the USSR Council of Ministers concerning the improvement of the economic mechanism in construction. The normative-reference base and software are being developed to carry out the engineering and economic preparation of the work. Passports are being compiled for objects-analogs that are formed by the computers on the basis of budgetary documentation, part IV of the norms of the Construction Standards and Regulations, YeRYeR [regional unified unit cost rates], YeR [expansion unknown], price lists, dictionaries and reference books.

It is necessary to note that in petroleum and gas industrial construction, in planning, and especially with the development of unitized sets construction, it is possible to use the concept of "object-analog" with a certain degree of error. This is the reservoirs, separators, automated measuring facilities, boiler flues, gas purification facilities, etc. After calculating on the computer according to the budgetary documentation and norms of the Construction Standards and Regulations the need for the resources to build these facilities, and after making the necessary refinements according to the working drawings and other project-estimate documentation, selected data will be able to be adopted as the normative-reference information for the objects-analogs.

For linearly extended facilities, external networks and communications, the set of the amount of work and resources for performing it for a unit of expanse of these facilities is taken as the analog, while for a shop of compressor and pump stations--for one aggregate of a specific type (capacity, performance).

In the process of forming passports for the facilities, the computer makes it possible to consolidate unit work, to make an automatic distribution of the work and resources selected from the Construction Standards and Regulations by performers with regard to established specialization, to make an element-by-element breakdown of direct outlays (i.e. the possibility emerges of singling out the cost of physical resources from the total cost of the work), to determine the full cost of the work with regard to plan accumulations and factors of limited outlays, and to determine the estimated (normative) labor-intensity and wage fund both for consolidated types of work and for the facility as a whole. The set of objects-analogs may correspond to an advance startup complex, the project as a whole, a program of work for a plan period, etc.

In addition, the computer is used to form facility-by-facility network (linear) schedules of construction and installation work. After correlating these schedules to the established deadlines for completion of construction of the facilities and combining them into a single schedule for construction, the computer is used to make a distribution of the amount of work and resources for performing it by quarters of the planned year and to check the balance of the plan with the resources with output of some recommendations on necessary changes in production capacities of the construction organizations and suspenses for performing the work for commissioning the production capacities and facilities on time.

Thus, with the help of the computer, a consolidated priority schedule is derived for commissioning production capacities and facilities with a distribution of work by performers and deadlines for delivery of materials and basic equipment. Based on the calculated data, for each performer is built a long-term brigade flow with regard to transfer from facility to facility. With the data on the analog facilities stored in the computer as normative-reference information, based on the title lists of the projects that are now an invariable plan document, one can computer the need for materials, equipment and manpower by performers for the entire period of construction as well as the need for resources to fit out specific deposits.

Construction industry workers have to learn how to use the developed system and with the aid of the computer prepare definitive solutions, specifying different versions of input information to obtain alternatives of possible solutions. The psychological barrier, which still exists, in our view, in many production workers, with respect to automated management systems, has to be overcome so that they become active users of the computers rather than just observers and critics.

We would like to touch on the questions of cost-reduction, stabilization and in a number of cases even reduction of the number of workers on the management staff when the computer is used in major construction organizations.

The reduction, carried out annually, in pay costs for administrative and managerial personnel in construction organizations on the scale of four to six percent for all practical purposes is not achieving its goal. The limited allocations, introduced several years ago, for pay of management staff are also not yielding the expected result since with the introduction of them, the management staff has essentially not been reduced, but the construction organizations have been permitting overruns and sometimes significantly exceeding the established limits.

In our view, there is now a real possibility of reducing the cost of the management staff and the number of workers on it. This is what has to be done to accomplish this:

transfer as much as possible all procedures for calculating technical and economic indicators of plan activity of a construction organization to a computer, after reinforcing this by a special decision;

organize the flow of information for reporting only through information-computer centers with orientation to making use of the existing pool of ASU hardware. With that, by-passing the IVTs to send any report and information to higher authorities has to be categorically prohibited;

using the computer hardware and communication facilities, organize centralized acquisition, storage and presentation of information in any combination of it for management decision-making;

develop and use strict, scientifically substantiated normative provisions, confirmed by practice, for the number of management staff personnel of organizations; and increase the responsibility of organization directors for, and stir their interest in, proper use of computer hardware and complying with the normative provisions stated above.

This conclusion can be drawn from what has been said: To raise the efficiency of introduction of computers and mathematical methods in the construction industry, it is necessary to extensively enlist workers at all levels of the management staff of a construction organization to create and maintain a centralized information-reference bank and to perform computing operations with the aid of the computer and data teleprocessing facilities.

In the Summary Report by the CPSU Central Committee to the 26th party congress, it was pointed out that a seemingly simple and very prosaic cause--a proprietary attitude toward social property and the ability to fully and expediently make use of everything that we have--has become the core of economic policy. To achieve this is an important task in all sectors of economic practice.

8545
CSO: 1863/220

UDC 621.73:621.7.077

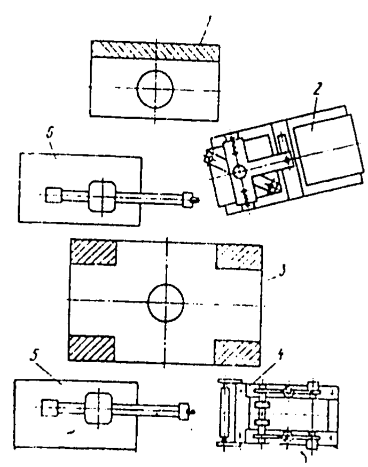
AUTOMATION OF MULTI-PASS SHEET-METAL STAMPING USING INDUSTRIAL ROBOTS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 6, Jun 81 pp 25-26

[Article by Doctor of Technical Sciences Ye. I. Semenov and Candidates of Technical Sciences S. A. Skorodumov, M. A. Kryuchkov and B. Suleymanov]

[Text] Analysis of the advisability of using industrial robots in sheet-metal stamping shows that one of the most effective areas of their application is the automation of multi-pass stamping on single-position presses, which are widespread at the present time in sheet-metal stamping shops [1-3].

Among the main factors affecting the arrangement of robot engineering complexes for sheet-metal stamping and the composition of their equipment are the technology of production of articles, the production rate of the complex and the possibilities of production areas.



The figure shows a variant of the arrangement of a robot engineering complex meant for the multi-pass stamping of the casing of the reflector of an electric train lamp. This robot engineering complex was developed in the All-Union Correspondence Mechanical Engineering Institute (Moscow) and made up according to a technological principle with installation of the equipment in the sequence in which the technological passes are made. It contains a twin-crank double-action press 3 with a force of 1000/630 kH, accomplishing the first operation of parts manufacture, a sheet-metal feeder 4 which feeds sheet-metal blanks lubricated on both sides from a packet into the position in which the blank is seized by the hand of an industrial robot; a single-crank press 1 with a force of 1000 kH which accomplishes the secon operation of parts manufacture; a stacker of parts 2 which stacks finished parts and two industrial robots 5 and 6, which service the presses and auxiliary transport and orienting equipment.

The work of all the robot engineering complex equipment is accomplished in the following sequence: the industrial robot 5 seizes the sheet-metal blank (420 x 330 x x 1 mm) lubricated on both sides, issued into the position of seizure by the sheet-metal feeder 4, transfers and places it in the die of double-action press 3, and then industrial robot 5 again moves into the position of seizure of the next sheet-metal blank, and the press performs the first operation, drawing. After the drawing of the part on press 3 has been completed in the upper position of its sliding bearings the blank is ejected from the matrix and the hand of industrial robot 6 grasps it and with a circular motion transfers and places the blank in the die of press 1, where the second operation is performed--holes are punched in the blank. After that operation has been performed and the blank is ejected from the matrix the industrial robot 6 transfers it to the parts storage 2 and sets it on the clasps of the latter. The stacker stores the blanks in trays.

Other variants of arrangement of the equipment and the use of single industrial robot and of a press with interchangeable dies are possible.

If the stamping is accomplished without oiling the blanks or they are oiled by special equipment, for example, jet spray nozzles, set in the stamping zone, one can use instead of sheet-metal feeders stackers of parts or devices for feeding sheet-metal blanks into the position where they are seized by the industrial robots.

The sheet-metal stacker is designed so that after a package of sheet-metal blanks has been loaded on the motor-operated table the feeding rollers are pressed against the upper sheet of the stack of blanks. Upon command coming from their drive mechanism the rollers start to rotate and, interacting with the upper sheet as a result of forces of friction which arise between the roller surface and the sheet, force the latter into the slot between the support strip and the strip of the equipment for limiting the number of fed blanks, and then between the lubricating rollers. Here the sheet is lubricated on both sides and is set in the position of seizure by a robot. The diameter of the feeding rollers and the gear ratio of the rotary mechanism of the feeding roller were selected so that they assure the pushing of the sheet through the slot during one turn of the feeding roller.

With increased height of the package of sheets, upon the signal of the corresponding sensors the drive mechanism for elevating the sheet feeder table is automatically switched on; with it the table with the package of sheets is automatically set in the starting position.

The work of the parts stacker during stacking of intermediate blanks in trays consists in this, that after the stamping the industrial robots transfer the blanks to horizontal shelves of rotating levers. During vertical movement below a special cross-piece, vacuum or magnetic clasps contact the blank and the rotating levers are withdrawn. The blank is held by the clasp and stacked on the tray and the stacker is also returned to the initial position upon issuance of a command from the sensors.

Intermediate blanks can be fed to the industrial robots and later transferred to the stamp tool by the parts stacker.

To set the filled tray a device for rotation of the load-bearing plate is provided, on which there is a special cross-section. During movement of the cross-section with vacuum or magnetic clasps the blank is drawn off the tray, transported vertically and set on the horizontal shelves of the parts stacker rotating levers. The blank is moved further by the robot.

When necessary the parts stacker can assure feeding of blanks from the tray in a single pass, the reception from an industrial robot of blanks of the next pass and their transport with a 180° turn upon the command of control for stacking in another tray.

Industrial robots of robot engineering complexes are equipped with clasps which assure the possibility of grasping only one sheet-metal blank and a system for monitoring the grasped blanks. All the robot engineering complex equipment is provided emergency blocking devices which eliminate breakdown of the main or auxiliary equipment when malfunctions arise in the work of equipment. The complex is successfully functioning at the Riga Car Building Plant.

The main functional units of robot engineering complexes are being demonstrated in the Mechanical Engineering Pavilion of the Exhibition of Achievements of the National Economy of the USSR.

BIBLIOGRAPHY

1. Noritsyn, I. A., and Vlasov, V. I. Avtomatizatsiya i mekhanizatsiya tekhnologicheskikh protsessov kovki i shtampovki (Automation and Mechanization of Technological Processes of Forging and Stamping). Moscow, Mashinostroyeniye, 1965.

2. Rovinskiy, G. N. Pressovoye oborudovaniye listoshtampovochnykh tsekhov (Press Equipment of Sheet-Metal Stamping Shops). Moscow, Mashinostroyeniye, 1960.

3. Belyanin, P. N. Promyshlennyye roboty (Industrial Robots). Moscow, Mashinostroyeniye, 1975.

2174
CSO: 8144/1447

UDC 621.311-51:681.3

USING COMPUTERS IN POWER SYSTEMS: FROM EXPERIENCE OF DONBASS POWER SYSTEM

Moscow PRIMENENIYE EVM V ENERGOSISTEMAKH: IZ OPYTA DONBASSENERGO in Russian 1981 (signed to press 16 Dec 80) pp 2-5, 181-182

[Annotation, preface and table of contents from book "Use of Computers in Power Systems: From the Experience of the Donbass Power System", by Il'ya Osipovich Kneller, Energoizdat, 8000 copies, 184 pages]

[Text] In this book, the author covers the questions of using computers for real-time supervisory and organizational-economic control of a power system. He discusses the basic stages of introduction of computers in power systems; the questions of acquisition, processing and transmission of information are considered; he describes the basic problems solved by using computers. Considerable attention is paid to questions of organization of operation of computer centers and subscriber stations in power systems. Technical-economic substantiation of use of computers is illustrated with examples.

This book is intended for power system engineers and technicians, and for design and scientific research organizations engaged in development of ASU's in power engineering.

Preface

Modern power engineering is a complex system. It is characterized not only by the presence of a large number of elements making up the system with multicircuit direct and feedback couplings, but also by certain special features: continuous generation of power and the complex nonlinear dependency of the basic technical-economic indicators on the process of generation and distribution of power and heat. Power generation and distribution depend to a considerable extent on weather, time-zone and other conditions.

The Donbass power system is one of the largest in the Soviet Union. The latest domestic power equipment is being assimilated and operating in the power system: pilot models of the 800,000 kW units at the Slavyanskaya and Uglegorskaya GRES's, the 800 kV DC Donbass-Volgograd power transmission system and the 750 kV AC Donbass-Western Ukraine power transmission system.

The electrical networks of the power system cover a territory of over 600,000 km^2 and provide a centralized supply of power to industry, railroads, agriculture and the population of Donetskaya and Voroshilovgradskaya oblasts.

It would be impossible to control this complex and large power system today without using cybernetic methods. A major area in the efforts to improve power system control over the last decade has been the extensive use of computers.

Due to the complexity of the Donbass Power System as a controlled object, without computers it would be impossible to solve the problems of optimal distribution of resistive loads between power stations, to make an economic selection of operating units, to compute the steady state of flow distribution in the main network, etc.

Realization of a number of optimizational calculations, the performance of which is possible only with a computer, yields a substantial economic effect.

The amount of organizational-economic information is continuously growing, especially under the conditions of centralization of control functions. Efficient processing of the large number of indicators (over a billion per year) would be impossible without the extensive use of computers, informational and managerial hardware.

In the Donbass Power System, where introduction of computers began in 1964, they are used in three main areas.

1. A computer is used for real-time supervisory control. In the first stage, computers were used for individual calculations of steady and emergency states. Later, programs for optimizational calculations were introduced, an information computing system for electrical calculations was established, and cathode ray tubes (CRT's) were introduced for output and input of information. Development of an automated system for supervisory control (ASDU) with automatic input of information and management of optimal states in real time is the next task in this area.

2. Computers are used to improve organizational-economic control. A brief list of this work: definition of plans for generating electricity and heat, analysis of sale and cost of power, accounting and distribution of physical resources and personnel, mechanization of accounting and calculation operations, development of information computing and information retrieval systems for the generating activity of the power system. Full mechanization and automation of all computing work under the conditions of centralization of the functions of power system control is the most important task in the current stage of improving the control of organizational-economic activity.

3. Computers are used to control the process of the large-capacity power units. The information computing systems that control the 800 MW units at the Slavyanskaya and Uglegorskaya GRES's effect monitoring of the technological parameters and calculation of the technical-economic indicators of operation of the units and carry out a number of functions for monitoring pre-emergency and emergency operations. This monitoring and calculations are performed in real time and create the prerequisites for maintaining an economic mode of operation of the units.

The successful introduction of computers in the power system was based on the works on theory and practice of cybernetic control of a number of scientific research, design and training institutes [1-6, 27].

At the end of 1974, more than 200 programs were in operation in the power system to solve problems of various classes on nine second-generation computers installed in the computer center (VTs) for the power system and in the facilities.

The new stage of introduction of computers, begun in 1975, is characterized by third-generation computers being put into service, the systems approach to solving a number of problems with a common data bank, the development of complexes of interrelated computers operating in the real-time mode, and by the development of an automated control system (ASU).

The purpose of this book is to bring out the experience of the Donbass Power System in using computers to solve problems of real-time supervisory and organizational-economic control. The author set himself the task based on the experience of one power system to illustrate the stages of introduction of computers; the development of an information base and system for acquisition, transmission and processing of information; operation of computers and solving a number of problems with them; and the effectiveness of using computers.

A considerable portion of the material presented reflects the results of a multiyear effort by the associates of the department of the ASU for the Donbass Power System and the author is grateful to V. R. Drachev, Ye. A. Kardasheva, A. I. Podol'skiy and others for assistance rendered in preparing material for the book.

Please send comments and suggestions on the book to: Izdatel'stvo "Energiya", 113114, Moscow, M-114, Shlyuzovaya nab., 10.

The author

Table of Contents

8545
CSO: 1863/235

UDC 621.311-51:681.3

AUTOMATED POWER CONTROL SYSTEM WITH THIRD-GENERATION COMPUTERS

Moscow PRIMENENIYE EVM V ENERGOSISTEMAKH: IZ OPYTA DONBASSENERGO in Russian 1981 (signed to press 16 Dec 80) pp 16-18

[Section 1-3 from book "Use of Computers in Power Systems: From the Experience of the Donbass Power System", by Il'ya Osipovich Kneller, Energoizdat, 8000 copies, 184 pages]

[Text] 1-3. Development of Automated Control System Based on Third-Generation Computers

In 1975, the M-4030 and M-6000 third-generation computers were installed in the power system and development of the power system ASU began. The main features of this stage:

1) establishment of the multimachine computer complex made up of three M-6000 computers (set numbers 2, 8 and 9), the M-4030 and the YeS-1040 computers; implementation of intermachine exchange of information; diagram of the information computer complex of the ASDU [automated supervisory control system] for the Donbass Power System is shown in fig. 1-3;

2) operation of the M-6000 computers in the real-time mode with input of telemechanical information from power system facilities;

3) development and implementation of the IVK [information computer complexes] (in real-time mode) for real-time supervisory control of the power system; CRT's (alphameric and graphic) are used extensively in these complexes for information display;

4) introduction of specialized and general-purpose data banks, development of information support system; introduction of the computer collective-use system;

5) computer operation in the multiprogramming mode using the ASVT DOS, YeS DOS and YeS OS (M-4030) operating systems; FORTRAN, COBOL, PL and other algorithmic languages are widely used;

6) development of information acquisition system using data transmission equipment (APD) and subscriber stations (AP); direct input of information from APD (without perforated tape) into computers;

7) broad introduction of problems of organizational-economic control; importance of these problems grows since a number of control functions are centralized in connection with the conversion (in 1976) of the regional power administration into a power generation association (PEO); establishment of main centers (see section 5-2).

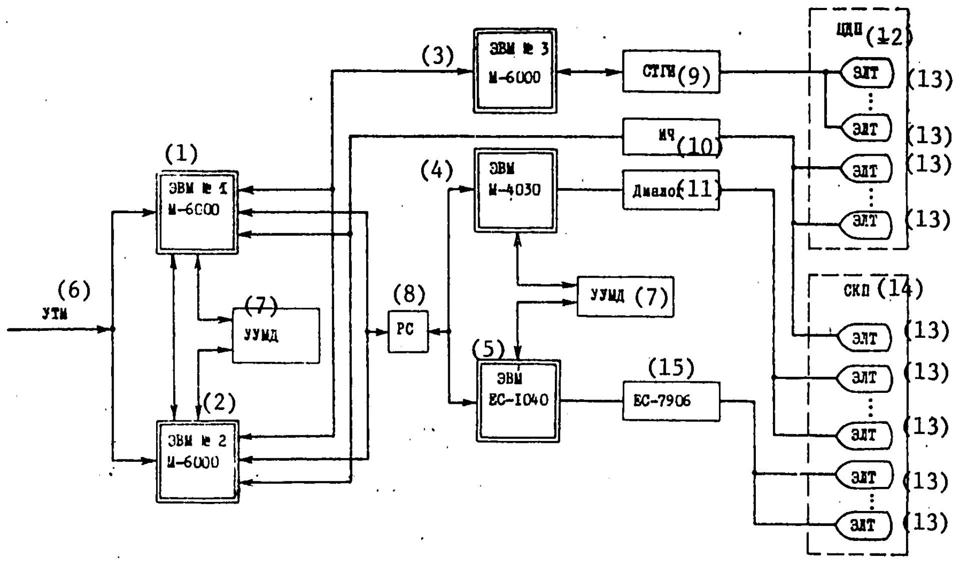


Fig. 1-3. Diagram of information computer complex of automated supervisory control control system for the Donbass Power System

Key:
1. M-6000 computer No. 1 with standard set No. 2
2. M-6000 computer No. 2 with standard set No. 8
3. M-6000 computer No. 3 with standard set No. 9
4. M-4030 computer
5. YeS-1040 computer
6. UTM -- telemechanical equipment
7. UUMD -- magnetic disk storage control unit
8. RS -- interface splitter
9. STGI -- television graphic display system with five displays
10. ICh -- supervisory information display unit
11. Dialog -- CRT information display system
12. TsPD -- central supervisory station
13. CRT
14. SKP -- collective-use system
15. YeS-7906

8545
CSO: 1863/235

UDC 621.311-51:681.3

ECONOMIC EFFECTIVENESS OF POWER SYSTEM AUTOMATED CONTROL SYSTEM

Moscow PRIMENENIYE EVM V ENERGOSISTEMAKH: IZ OPYTA DONBASSENERGO in Russian 1981 (signed to press 16 Dec 80) pp 174-177

[Appendix from book "Use of Computers in Power Systems: From the Experience of the Donbass Power System", by Il'ya Osipovich Kneller, Energoizdat, 8000 copies, 184 pages]

[Text] Appendix

Effectiveness of Automated Power Control System

The quantitative values of the economic effect achievable in solving optimizational problems may be derived by using an algorithm, the essence of which consists in the following: Power system operating conditions have to be modeled twice for a common period of time (T) with a computer--once for manually maintained conditions and once for those maintained with computers; then the economic characteristics of these models are compared. For the problem of optimizing distribution of resistive loads among power stations, this algorithm consists in the following.

Using the computer, the total fuel consumption for the power system (V_{actual}) is calculated for the period of time T on the basis of the actual load curves for the power station units and the corresponding KhOP [incremental rate characteristic] of the operating composition of equipment (when conditions are maintained manually).

For this same period T, the computer is used to perform an optimal distribution of power system curves among the power stations and fuel consumption $V_{optimum}$ is defined. This calculation is made for the same incremental rate characteristic and composition of units, and then the difference is determined

$$\Delta V = V_{actual} - V_{optimum}.$$

This difference ΔV is also the maximum possible savings in fuel obtainable in solving the problem.

In the Donbass Power System, computer calculations were made for 60 days in 1974 using this algorithm and the following result was obtained: fuel savings were 0.3 percent or 1,113,000 rubles per year. In the process, the assumption was made that the initial values of all factors coincide with real values, and the control actions (optimal power station load curves) derived in the calculation are realized without deviations. In reality, the error in determining the initial factors and realizing control actions may be very substantial. Quantitative estimates of the effect of optimizing power system conditions when errors are present may be derived through statistical evaluation of the results of the cybernetic modeling of the processes of movement and processing of information [3].

As a result of statistical calculations to estimate the effect of accuracy of input information and degree of realization of calculated optimal curves on the optimization function--fuel consumption in this problem, the value of the economic effect was reduced to 0.23 percent or 790,000 rubles per year.

A similar algorithm was used to calculate the economic effectiveness of the IVK [information computer complex] "Optimization and Calculation of GRES Technical-Economic Indicators." In the mode "Actual" of this complex, fuel consumption V_{actual} was determined for past periods according to the actual load curves and power characteristics of the units (see section 1-2). For these same periods, in the mode "Optimum," fuel consumption $V_{optimum}$ was calculated with optimal distribution of the actual curve of the power station loads among its units. The maximum possible fuel savings was derived as the difference between actual and optimum fuel consumption. The economic effect was determined using one GRES as an example. Statistical processing of the calculations made daily throughout 1974 in the modes "Actual" and "Optimum" showed that the maximum effect corresponds to a 1.3-percent reduction in fuel consumption equivalent to 450,000 rubles per year. Due to a number of technical limitations (shutdown of mill systems, slagging of boilers, etc.), the optimal curves could not have been fully realized; therefore, this economic effect has to be reduced. To identify the economic effect reduction value, an effort was made to determine fuel consumption reduction dependency on the degree of realization of the optimal curves. Calculation results showed that this function is linear. Realization of optimal GRES curves is no more than 45 percent, which reduces the IVK economic effect to 0.58 percent of annual fuel consumption. This complex of programs was introduced in three power stations. To analyze the economy of the optimal conditions for operation of the Donbass Power System high voltage networks, a series of calculations were made as follows. Optimal transformation ratios were defined according to the program for optimization for one of the conditions (winter peak). Then, on the assumption that these transformation ratios are shown in the corresponding facilities, calculations were made for a series of balances to compare the power loss before and after optimization of the conditions. The average reduction in loss for 14 conditions (9.4 + j 143.4 MVA, or 4.5 percent) shows that substantial savings can be obtained for numerous different conditions even in the case of establishing under different conditions the same transformation ratios (optimal for one of the balances). The total savings for the power system through power loss reduction in the main network was 280,400 rubles per year.

Calculations on selecting optimal schemes in distribution networks have been made over a number of years in the power system. Calculations for 36 Donbass cities showed that power losses can be reduced 5 to 30 percent thanks to optimization. For example, in Zhdanov's distribution network, power losses in the calculation on the original network were P_{actual} = 1607.4 kW, while losses in the optimal scheme were $P_{optimum}$ = 1086.7 kW. Thus, loss reduction in the network was 520.7 kW.

With a period of use of highest load by consumers of $T_{highest\ load}$ = 6000 h and cos = 0.9, loss time = 4300 h and power savings through loss reduction was 520.7 X 4300 = 2,239,010 kWh.

Table A1 shows the values of annual economic effect from introduction of the program for selecting optimal schemes in distribution networks for 36 Donbass cities.

Table A1

Year	Savings from optimization, rubles	Outlays for computer calculations, rubles	Annual savings, rubles
1973	177,100	5,900	171,200
1974	158,100	5,900	152,200
1975	42,900	1,900	41,000
Total	378,100	13,700	364,400

Let us now consider the effectiveness of an organizational-economic problem. Of importance from the point of view of mechanization is the problem of maintaining customer accounts for power sales. In the Donbass Power System, there are up to 2 million customers which on the average make up to 20 million payments per year. To maintain the accounts and monitor payments coming in, the power sales division has large staffs of accounting workers and supervisors who manually sort receipts, make various spot checks and financial calculations, and determine the completeness and timeliness of payments for power. It is quite natural to wish to ease the low-output manual labor by taking advantage of the large pool of modern computer equipment. The question arises of what equipment to choose and just what the degree of mechanization should be to make it efficient for processing the large number of payments coming in. Besides the traditional manual method, there are two possibilities for solving this problem with punchcard equipment and with computers.

Let us briefly describe each information processing scheme and make a technical and economic comparison of them.

The Punchcard Machine (SPM) Processing Scheme. The receipts paid by customers at savings banks come into the computer center's information receiving group, are recorded and then sent to the punchcard group. Checking for correctness of punching is done by verification of the punchcards and counting checks.

The checked files of cards are sorted on an electronic sorter and fed through a tabulator with the VP-3 calculating attachment. As a result of the processing of the power sales receipts, the following printouts are made: "Arrival of Payments by Savings Banks and Transfer Dates," "Distribution of Payments by Personal Accounts," "Results of Work of Supervisors for the Month," and others.

Computer Processing of Receipts after They Are Punched. Put on computer magnetic tape (ML) for each customer are the personal account number, rate, meter number and reading and the address. Also recorded in the personal account are the meter readings paid for and payments received for the last 12 months. Receipts paid by customers are punched on cards, verified and sent to the computer for processing. The cards are loaded on magnetic tape as they come in during the entire day; at the end of the day, a report is put out on payments received by savings banks and transfer dates. The payments recorded on tape are sorted by account number and posted to customer personal accounts. Upon request, printouts may be made on: payments received by savings banks for any transfer date, work indicators for power sales sections, information on accounts not paid, results of supervisor's rounds, payments by events, as well as personal accounts. This algorithm has been implemented on the M-220 computer.

Let us compare the effectiveness of maintaining accounts by the methods described using an analysis of outlays for processing of a document.

Calculations for the technical and economic comparison of the information processing methods for customer accounts were made using the example of the Gorlovskoye Department of Power Sales with payments received of one million receipts per year. According to the procedure developed by the TsNIITU [Central Scientific Research and Technological design Institute for the Organization and Technology of Control], the total cost of outlays for processing information is calculated by the formula

$$C_{tot} = C_{wage} + C_{amort} + C_{elec} + C_{rep} + C_{mat} + C_{aux} + C_{other}$$

where C_{wage} is the fund for fixed and supplementary wages of personnel allowing for social insurance deductions; C_{amort} is total deductions for amortization; C_{elec} is cost of power consumed by equipment; C_{rep} is cost of spare parts; C_{mat} is cost of materials; C_{aux} are outlays associated with operation of auxiliary equipment and implements; and C_{other} are other expenses.

Consolidated in table A2 are the results of calculation of outlays for manual, punchcard and computer processing. Thus, with the algorithms described for organization of calculations, using the M-220 computer for customer accounts is not effective; the method of processing receipts using punchcard machines turns out to be more economical.

Table A2

Indicators	Processing Method		
	Manual	Punchcard	Computer
Wage fund, rubles	22,032	14,076	7,718
Amortization deductions, rubles	--	1,286	494
Cost of power consumed, rubles	--	314	174
Cost of spare parts, rubles	--	1,439	1,003
Cost of materials, rubles	--	959	400
Outlays for auxiliary equipment, rubles	--	479	250
Other expenses, rubles	--	95	95
Outlays for calculation by computer, rubles	--	--	20,200
Total outlays, rubles	22,032	18,648	30,334
Outlays per receipt, kopecks	2.2	1.8	3.03

8545
CSO: 1863/235

PUBLICATIONS

ABSTRACTS FROM JOURNAL 'AUTOMATIC CONTROL AND COMPUTER SCIENCE', MAY-JUNE 1981

Riga AVTOMATIKA I VYCHISLITEL'NAYA TEKHNIKA in Russian No 3, May-Jun 81 pp 97,99

UDC 681.324

INTERNATIONAL INFORMATION SERVICE BASED ON COMPUTER NETWORKS

[Abstract of article by Kh. M. Khinov, A. V. Butrimenko, V. N. Kalachev and A. Ye. Umnov]

[Text] The international information service made available by distrubuted data banks based on computer networks is a rapidly growing type of information services. Its export is a very unique form of international trade in which the sale of the commodity, information and data processing services in this case, does not reduce the supply of it in the exporting country and does not require any nonrenewable national resources at all. The major features of the data banks and computer networks are described. An analysis is made of the current status and prospects for development of the international market for information services. Discussed are the organizational-legal questions of setting up and operating international computer networks which are being worked out intensively and for which there is no historical experience. Bibl. of 9 titles, 3 figs.

UDC 681.324

DEADLOCK DETECTION DURING INTERACTION OF INFORMATION PROCESSES IN COMPUTER NETWORKS

[Abstract of article by E. V. Zinov'yev and A. A. Strekalev]

[Text] A model of distribution of data elements of processes functioning in parallel based on Petri nets is discussed. A method is suggested to detect deadlock groups based on computation of the matrix rank of the deadlock of blocked processes. Bibl. of 5 titles, 7 figs.

UDC 681.32:519.713

DESIGN OF DISCRETE INDUSTRIAL CONTROL DEVICES BASED ON PROGRAMMABLE LOGIC ARRAYS

[Abstract of article by G. Franke and V. P. Chapenko]

[Text] Discussed are different structural realizations with PLM [programmable logic arrays = PLA] of control devices resistant to races and chatter of input signals.

Standard realizations with PLA meeting these requirements are suggested. Three parts of PLA are singled out: the part representing the conditions of transitions in the device being designed, the part reflecting the conditions of stability of the device and the part reflecting the conditions of issuance of output signals. In PLA structure synthesis, the corresponding PLA parts have to be filled in as a function of the selected structural realization by the standard form in accordance with the control graph that describes the device being synthesized. The suggested method requires few outlays for synthesis. Questions of optimization of the PLA structure in the process of synthesis are not discussed. Bibl. of 9 titles, 6 figs., 3 tables.

UDC 519:7

NEGATIVE EFFECTS OF ELIMINATING UNESSENTIAL VARIABLES

[Abstract of article by A. A. Sapozhenko and L. M. Karakhanyan]

[Text] It is proven that eliminating unessential variables with minimization of partial Boolean functions can lead to exponential growth of the complexity of the DNF [disjunctive normal forms] derived. Using Akers' algorithm as an example, it is shown that heuristic algorithms of minimization that use the operation of eliminating unessential variables may obtain DNF significantly different in complexity from the minimal as well as DNF that are not deadlock. Bibl. of 9 titles, 7 figs., 1 table.

UDC 519.713

COMBINED IMPLEMENTATION OF SWITCHING FUNCTION SYSTEMS BY A PROGRAMMABLE LOGIC ARRAY CIRCUIT

[Abstract of article by Ye. S. Bul']

[Text] Suggested is a technique of synthesis of a circuit of programmable logic arrays (PLA) that implements the system of switching functions, specified in the set of compositions, as a function of the number of arguments that exceed the number of PLA inputs. Discussed is a method of determining subfunctions with decomposition of the initial system for selected breakdown of the set of arguments. An algorithm for selecting the breakdown of the set of arguments is suggested that allows obtaining a circuit of the minimal or close to minimal number of PLA through decreasing the multiplicity of the combined decomposition of the functions of the system. Bibl. of 8 titles, 6 figs., 3 tables.

UDC 519.711.7:681.324

GLOBAL OPTIMAL STRUCTURAL SYNTHESIS

[Abstract of article by D. I. Mladenov and T. A. Stoilov]

[Text] Discussed are questions associated with global optimal structural synthesis of a data transmission network. The suggested approach has a more general nature. It is shown that the optimal solution to global structural synthesis cannot be derived using existing solutions to partial problems. The single extremal problem cannot be solved in view of the mutual effect between these problems. These effects are investigated and a general problem of global optimal structural synthesis is suggested that is formed by the composition of partial problems and which considers the effect between them. The solution to it is given based on the theory of coordination using the solution to the partial problems. The synthesis is effected by a single computational logic procedure. Bibl. of 11 titles, 2 figs.

UDC 681.325.6

CALCULATION OF DISCRETE DEVICE FUNCTIONING VALIDITY TAKING PASSIVE MARGIN OF MONITORING TIME INTO ACCOUNT

[Abstract of article by A. M. Romankevich and A. D. Gorozhin]

[Text] Formulas are considered that allow raising the accuracy of calculation of the probability of detecting malfunctions of discrete devices with monitoring, that have a passive margin of monitoring time. Bibl. of 1 title.

UDC 681.324:681.3-181.48

MULTIMICROPROCESSORS IN PACKET SWITCHING NETWORKS

[Abstract of article by J. Sexton]

[Text] This article covers the problems of using microprocessor systems in packet data transmission networks. Discussed are the trends in development of these systems, new approaches in solving the problems of systems architecture, organization of storage, ensuring reliability, flexibility, restoration of operating capability, etc. Examples of specific systems are given.

UDC 681.324

TRANSPORT FUNCTIONS IN X.25 NETWORK

[Abstract of article by M. Zibert, Yu. S. Podvysotskiy and S. V. Rotanov]

[Text] Functions of the transport level of a system built on the base of the MKKTT [International Telegraph and Telephone Consultative Committee] recommendation X.25 are discussed. Discussed are problems of choosing a list of performable functions based on the relationship between the service offered by the network level and the service required of the transport level. Bibl. of 9 titles, 7 figs.

UDC 621.317.755

STROBOSCOPIC CONVERSION OF PULSE SIGNALS WITH RANDOM PHASE

[Abstract of article by V. G. Karklin'sh and K. Ya. Krumin'sh]

[Text] Analyzed are nonlinear distortions that occur in stroboscopic coversion of square pulse signals in the case when the phase of the strobe pulse relative to the signal being converted is unstable. A comparative analysis is made of the distortions characteristic for analog and discrete stroboscopic converters; ways of eliminating or minimizing them are given. Bibl. of 2 titles, figs. 4.

UDC 519.246

DETERMINATION OF STATISTICAL CHARACTERISTICS OF RANDOM TIME INTERVALS

[Abstract of article by D. V. Kutsevalov]

[Text] The author solves the problem of determining moments of distribution of random magnitude by moments of sampling distribution when the sampling values cannot be precisely measured. He derives an expression to determine the moments of distribution of an arbitrary order that is applicable to a broad class of distributions. Bibl. of 2 titles, 1 fig.

UDC 681.325.5-181.4

ORGANIZATION OF COMPUTER-INSTRUMENT INTERACTION IN SYSTEMS FOR AUTOMATION OF SCIENTIFIC RESEARCH WITH VARIABLE STRUCTURE

[Abstract of article by S. N. Domaratskiy, O. S. Zudin and O. Vaynio]

[Text] The authors consider questions of organization of computer-instrument interaction in laboratory systems for automation of scientific research that form single-center computer networks of the scientific research ships of the USSR Academy of Sciences. They introduce the concept of the process of conducting an automated experiment and consider the interaction of processes occurring in the different instruments participating in the experiment. They point out the advantages of using standard interfaces; of these, special attention is paid to the MEK [International Electrotechnical Commission = IEC] instrument bus. They give the structure, description and example of use of the universal interface card of the IEC bus, built on the base of a microprocessor and specialized microcircuits of programmable logic and suitable for interface with the system of the overwhelming majority of instruments with parallel or serial input and (or) output of data that are used in scientific research automation systems. They consider questions of building programs that interpret functions of interface with the IEC bus of specific instruments through the universal interface card. They describe the methods used to organize laboratory systems for automation of scientific research on the multipurpose scientific research ship of the USSR Academy of Sciences, the "Akademik Mstislav Keldysh." Bibl. of 14 titles, 6 figs., 1 table.

UDC 519.512

CORRECTNESS EVALUATION OF EXPONENTIAL DISTRIBUTION USAGE IN CLOSED SYSTEMS ANALYSIS

[Abstract of article by V. D. Mal'shakov, A. R. Timoshenko and M. B. Tamarkin]

[Text] The authors discuss a two-phase closed queueing system that is a subsystem of a computer complex with two instruments A and B and two queries. With this example, it is shown that using exponential distributions achieves determination of optimal solutions by the composition of phases of the system of the given type. Bibl. of 5 titles, 1 fig.

8545
CSO: 1863/223

UDC 519.72

DISTRIBUTION OF REGULAR MESSAGE FLOWS IN INFORMATION SYSTEMS

Alma-Ata RASPREDELENIYE REGULYARNYKH POTOKOV SOOBSHCHENIY V INFORMATSIONNYKH SISTEMAKH in Russian 1980 (signed to press 19 Jun 80) pp 2-4

[Annotation and foreword from book "Distribution of Regular Message Flows in Information Systems", by Vil'zhan Mavlyutdinovich Amerbayev, Vladimir Ivanovich Vasil'yev, Igor' Mikhaylovich Gurevich and Ivan Timofeyevich Pak, Izdatel'stvo "Nauka" Kazakhskoy SSR, 1,000 copies, 144 pages]

[Text] Playing an important role in the creation of computer centers and the development of collective-use information systems is the distribution of regular message flows circulating within them together with random information flows. There has to date appeared no publication systematically presenting from a unified point of view the problems associated with the organization and distribution of regular message flows within information systems. The book here offered the reader to some extent fills this gap. It examines the distribution of regular, including periodic, message flows within communication-channel switching centers and messages in information-processing centers comprising large information systems. It presents examples of practical utilization of methods now developed.

The book is intended for scientists and engineers involved in the planning and design of information systems with regular message flows as well as for instructors, graduate students and upper-class students in corresponding VUZ departments.

19 illustrations, 1 table, 46 bibliography entries.

Foreword

Constituting one component of the infrastructure of contemporary society is a variety of systems for collecting, storing, transmitting and converting information. Because of the diversity and complexity of the problems arising in connection with the development, introduction and improvement of information systems, and in view of the urgent need to insure the effective operation of higher-level systems, there arises the need to define and solve a number of problems associated with the distribution within information systems of regular, or determinate, information flows.

The book poses and discusses the problem of determinate time distribution of regular flows in information systems. Information concerning the coordinates of an n-dimensional object flows periodically or in accordance with a number of fixed algorithms into one or several communication or processing channels (servicing units), which time-distribute the messages. The fact that simultaneous transmission of information concerning two or more coordinates is impossible requires solution of the problem of time-distribution of the information within the channel of communication, the establishment of time-distribution criteria in various situations and, by employing these criteria, organization of optimum transmission of the data.

The first chapter of the book provides a brief survey of the methods of multiplexing channels of communication, including time-distribution methods. It presents examples of information systems employing time multiplexing (distribution) of regular, or determinate message flows. It enumerates typical tasks in distributing regular message flows associated with construction of telemetry and computer systems as well as of data transmission systems.

The second chapter describes the most widely employed means of distributing message transmission periods based upon known methods of quantizing continuous messages. It contains expressions for calculating the message transmission periods providing the required precision.

Discussed in the third chapter are a number of methods of calculating message transmission duration. Some of these are based upon utilization of the procedure of optimum quantizing of continuous messages; others take into account the fact of the utilization of a single communication channel for transmission of messages from several sources of information. Employing the uncertainty relation as a limit, the authors calculate optimum fidelities and, thereby, transmission durations with exponential and quadratic loss criteria.

The fourth chapter is devoted to a discussion of methods of analyzing and distributing determinate (primarily periodic) message flows.

Chapter 5 describes the apparatus employed in study of the problems of distributing combined (determinate and stochastic) message flows in a channel of communication. It discusses the steady-condition and gives expressions for determining the vector describing the states of the system. It presents analyses of maximum and average queues, a simulation model and results of statistical modeling. M. A. Medrish assisted with preparation of material for this chapter.

The sixth chapter contains examples of the use of the determinate distribution methods presented in the book to calculate a number of information system characteristics. It gives expressions for calculating communications channel parameters (throughput, band width, signal power). Discussed also is the problem of optimum selection of message transmission durations. The chapter describes a procedure for making optimum selection of methods of employing information transmission and processing equipment.

8963
CSO: 1863/204-A

COMPUTERIZED AUTOMATIC TEACHING SYSTEMS

Minsk AVTOMATIZIROVANNYYE OBUCHAYUSHCHIYE SISTEMY NA BAZE EVM in Russian 1980 (signed to press 6 Jun 80) pp 2, 169-170, 175

[Annotation, appendix and table of contents from book "Computerized Automatic Teaching Systems", by A. F. Chernyavskiy (ed.), A. M. Mukharskiy, A. V. Orlov, L. V. Strikeleva, A. I. Tarnopol'skiy and N. I. Zelenkov, Izd-vo BGU im. V. I. Lenina, 2000 copies, 176 pages]

[Excerpts] This book discusses the problem of the pedagogical application of computers and various aspects of its solution using automatic teaching systems (ATS). On the basis of experience accumulated in the development and introduction of an experimental computerized teaching system (EVOS-BGU) [the experimental computerized teaching system of the Order of the Red Banner of Labor Belorussian State University imeni V. I. Lenin] and analysis of existing foreign ATS it sets forth requirements imposed upon automatic teaching systems, principles governing the planning and design of the hardware involved and associated instructional methods and software. It presents results from the experimental operation of an automated EVOS classroom. It also indicates basic directions for further development of ATS.

The book is intended for specialists in the development and application of machines in the instructional process and in the professional training of staff personnel, methods specialists and instructors. It may also be useful to engineers and students involved in the planning and design of conversational systems and automatic control systems.

Appendix

Table 1. Foreign automated teaching systems

Name of teaching system, country	Year of installation	Computer	Type student position	Number positions	Remarks
CLASS (US)	1961	PHILCO-2000	movie screen, input console	200	
Stanford Univ. (US)	1963-69	PDP-10, PDP-1, PDP-8	teletypes, displays	200	
PLATO-II (US)	1962	ILLIAC-I	slides on television screen	2	
PLATO-III (US)	1964-1969	CDC-1604	television monitor, keyboard		16 hours of instructional material

Name of teaching system, country	Year of installation	Computer	Type student position	Number positions	Remarks
PLATO-IV (US)	1971	Cyber-170	Plasma panel, keyboard	up to 1008	2500 hours of instructional material in 60 disciplines
USA	1970	IBM-1130, PDP-8	16 displays		
KOS (US)	1970	IBM-360/75	Storage oscillograph, keyboard	up to 60	Complex numbers, electrical engineering
STRAND (US)	1971				
FOBOS (US)		ICL-1904A		up to 210	Mathematics in the school, Instruction in programming
TICCIT (US)	(1973)		Colored displays, special teletypes		Mathematics, English
Great Britain			Teletype, slide projector	15	Applied statistics, physical chemistry
O. P. E. (France)		IBM-360/30	Slide projector, typewriter	16	Self-checking in physics, English and chemistry
FRG	1974		Display	5	Self-checking, presentation of informational material
Great Britain		IBM-370/135	Display		Chemistry
US		PDP-10	Graphics displays	2	Organic chemistry. stereochemistry

Table 2. Automated teaching systems in the USSR

System	VUZ	Year of installation	Computer	Student position	Number work positions
RADON	KVIRTU	1962	Dnepr-1	Slide projector, toggle switches	16

System	VUZ	Year of installation	Computer	Student position	Number work positions
	MVTU	1968	Minsk-2	Indicator light, 10 keys	
MVS	Belgosuniversitet	1970	Minsk-2	teletypes	6
Repetitor	MEI	1971	Setun'	5 buttons	
Pedagog	Kiev National Economic Institute	1973	Dnepr-21	Teletype	7
	MIIT	1974	Ural-14D, Nairi	Teletype	
	MESI	1975	Minsk-32	Teletype	
	LETI		Minsk-22M	Teletype	up to 16
Student	Gor'kiy University		M-222	Keyboard	7
EVOS	Belgosuniversitet	1975	Minsk-32	Display screen consoles	16
SADKO	ZIL plant vtuz	1975	Minsk-32	Videoton-340, televisions	16
	MIFI	1976	Minsk-32	Videoton-340	8
ESPO	MESI	1976	YeS-1020	YeS-7906	16
ATOS	Belgosuniversitet	1979	YeS-1022	Special terminals	16

Key to acronyms in Table 2 (in order of appearance): KVIRTU - no expansion available; MVTU - Moscow Higher Technical School imeni N. E. Bauman; Belgosuniversitet - Belorussian State University imeni V. I. Lenin; MEI - Moscow Power Engineering Institute; MIIT - Moscow Institute of Railroad Transportation Engineers; MESI - Moscow Institute of Economics and Statistics; LETI - Leningrad Electrotechnical Institute imeni V. I. Ul'yanov; ZIL plant vtuz - ZIL plant higher technical educational institution; MIFI - - Moscow Engineering Physics Institute.

Table of Contents

8963
CSO: 1863/204

CONTROL SYSTEMS DESIGN AUTOMATION

Moscow AVTOMATIZATSIYA PROYEKTIROVANIYA SISTEM UPRAVLENIYA in Russian No 3, 1981 (signed to press 5 Feb 81) pp 2, 203-205

[Annotation and abstracts of articles from book "Control Systems Design Automation", issue 3 of a collection of articles, edited by V. A. Trapeznikov (chairman of the editorial board), I. V. Prangshivili, V. L. Epshteyn, A. G. Mamikonov, I. G. Dmitriyeva and D. M. Berkovich, based on papers presented at the All-Union Conference on ASU Design Automation held in Suzdal' in April 1979, Izdatel'stvo "Finansy i statistika", 12500 copies, 208 pages]

[Text] This collection covers questions of methodology of designing automated systems for control of production, and features of design and reliability of control systems software. In the collection of articles, the authors analyze the structural algorithmic approach and control of automation of systems of design based on full-scale, expert and model experiments.

This collection is intended for scientific research, design and educational institutions concerned with the problems of design automation.

METHODOLOGY OF DEVELOPMENT OF DESIGN AUTOMATION SYSTEM FOR AUTOMATED ENTERPRISE CONTROL SYSTEM

[Abstract of article by V. V. Yvdokimov]

Author discusses basic principles of system to automate design of automated enterprise control system (SAPR ASUP). He discusses methodological questions of constructing a generalized economic mathematical model of an enterprise management system and decomposition of it. He covers the basic principles of automating design of an ASUP data processing system and brings out the methodological questions of the technology of design of ASUP using SAPR facilities.

METHODS AND MEANS OF AUTOMATED DESIGN OF CONTROL SYSTEMS SOFTWARE IN THE YAUZA-6D AUTOMATED PROGRAMMING AND DEBUGGING SYSTEM

[Abstract of article by V. V. Lipayev, L. A. Serebrovskiy, F. A. Kaganov, B. A. Korepanov, M. A. Minayev and A. A. Shtrik]

Authors discuss general problems of automating design of complexes of control programs and means supporting technology and process of development. They give functional schemes of the YaUZA-6D automated programming and debugging system (SAPO) and systems included in it, brief description and technical characteristics of them, and structure of the set of documentation for the SAPO.

SEMIOTIC SYSTEMS OF DESIGN

[Abstract of article by Yu. I. Klykov]

Author considers principles of constructing semiotic design systems and ways of developing them. He analyzes reasons for weakness of design systems based on formal models. He shows need for developing semiotic design languages.

AUTOMATION OF DESIGN OF AUTOMATED PROCESS CONTROL SYSTEMS (AUTOMATED SYNTHESIS)

[Abstract of article by V. V. Solodovnikov]

Author deals with questions concerning one of the most complex stages in designing automated process cntrol systems that is aimed at satisfying specified requirements for dynamic properties and accuracy of the system.

He analyzes the features of the ASUTP design automation process and list the requirements which must be satisfied by the algorithm for automated synthesis.

He presents the principles of the frequency method of synthesis and theory of coordinated control that can be used in developing the composition of special software for automation of ASUTP design.

OPTIMIZATION IN AUTOMATED DESIGN OF CONTROL SYSTEMS

[Abstract of article by I. V. Safonov]

The author presents the experience of stating and solving optimization problems in designing algorithms and programs for automated control systems and data processing systems. He formulates a series of new statements of problems of optimization design of real-time algorithms and gives an example of solving one of these problems. He notes the need for interactive optimization in working out design solutions.

MODELS AND METHODS OF AUTOMATING DESIGN OF MODULAR DATA PROCESSING SYSTEMS

[Abstract of article by A. G. Mamikonov, A. A. Ashimov, V. V. Kul'ba, S. A. Kosyachnko and G. Z. Kaziyev]

Authors discuss models and methods of automating basic stages of design of modular data processing systems (SOD). They offer a general structural scheme of a system for automating design of a modular data processing system and the basic principles of its functioning. They consider the features of stating and solving problems of optimal synthesis of software and information support for SOD.

CONTROL SYSTEMS DESIGN AUTOMATION IN ORGANIZATIONS OF THE USSR MINISTRY OF HIGHER AND SECONDARY SPECIALIZED EDUCATION

[Abstract of article by I. M. Makarov]

Author describes status of developments on creating systems for automation of planning and design that are underway in the USSR Ministry of Higher and Secondary Specialized Education. He discusses the makeup of a standard center for automated design and engineering approved by the USSR Ministry of Higher and Secondary Specialized Education. He gives brief characteristics of program complexes oriented to solving specfific problems of control of concrete objects. He discusses prospects of creating automated teaching systems.

SOME EVALUATIONS OF EFFICIENCY OF PARALLEL COMPUTATIONS

[Abstract of article by E. A. Trakhtengerts]

Author considers capability of parallel computations with various types of multiprocessor computer systems and gives evaluations of efficiency of parallel computations for certain types of equations.

STRUCTURE OF DEVELOPMENTAL AID COMPLEX, AUTOMATED DEVELOPMENT OF INFORMATION MANAGEMENT SYSTEMS

[Abstract of article by V. G. Volin, V. A. Gruzman, V. I. Senichkin and V. L. Epshteyn]

Authors describe the functional algorithmic structure of the developmental aid complex ARIUS [automated development of information management systems] that implements procedures of analysis and design of information management systems. They discuss questions of automating analysis of information flows and building an integrated data base; they give informational design of special-purpose management system and fundamental scheme of its development.

SYSTEM CONCEPT OF THE MARS COMPUTER AIDED DESIGN SYSTEM FOR AUTOMATED CONTROL SYSTEMS

[Abstract of article by E. N. Khotyashov]

Author discusses a model of a hypothetical object (hypothetical model) that is the basis of the MARS system, and systems for development and running of the hypothetical model and model of the object. These models afford a choice of the hardware complex and data base structure and enable developing a data base management system and data processing programs in ASU's. He describes a method of modernizing a functioning ASU.

CHARACTERIZATION CONTROL IN CONSTRUCTING HIGH-SPEED COMPUTER AIDED DESIGN SYSTEMS

[Abstract of article by V. A. Gorbatov and N. V. Fedorov]

Authors discuss the problem of developing high-speed computer aided design systems, for the solution of which is used the principle of characterization control. This approach allows efficient reduction of the computational process in searching for optimal solutions which support development of high-quality digital equipment.

An illustration of the efficiency of characterization control is the "Control-21" automatic system for design of complex logic structures.

USE OF FORMULA LANGUAGE TO FORM DATA BASE AND SOLVE INFORMATION LOGIC PROBLEMS

[Abstract of article by Kh. K. Brutyan, L. V. Mkrtchyan and S. A. Sorokin]

Authors propose a program system belonging to the class of general-purpose systems to automate design of information support. System is intended for automation and design of data bases and oriented to solution of accounting, calculating, planning and other problems.

AUTOMATED EXECUTION OF DESIGN DOCUMENT

[Abstract of article by L. V. Sazonova and Yu. A. Sokolinskiy]

Authors discuss capability of automating execution and drawing up of design documents containing calculations and explanations for them through automatic generation of the calculation procedure while processing source text that specifies technique of calculation in input language. They describe the grammar of the language, algorithms for the generation program and document editing procedures. They present results derived for some subset of the input language.

STRUCTURAL ALGORITHMIC APPROACH TO CONTROL OF DESIGN SYSTEM AUTOMATION

[Abstract of article by A. S. Grinberg and Ya. A. Shabad]

Authors discuss approach to control of design system automation based on expert model and full-scale experiments. They suggest organizing these experiments in the form of a periodic competition of various alternative ASP [automated design systems] according to the results of which an evaluation of each ASP alternative is formed. In accordance with the evaluation, a choice is made of problems of design subject to automation in each ASP, and corresponding objects of design are assigned to the ASP. This control orients the development of the ASP to achieving end results.

USE OF SIMULATION IN AUTOMATED DESIGN SYSTEMS

[Abstract of article by G. G. Chogovadze]

Author discusses problems associated with use of simulation in automated design systems (SAPR). He suggests concrete solutions for SAPR, a design object of which are integrated control systems. A flow model of an enterprise is used in the process. He analyzes the problem of substantiating the technical specification for an ASU by results of a system survey and the problems of automating the system survey.

The design solutions for the problems in question allow constructing a technique of design for large program systems for the purposes of organizational and technological controls.

IDENTIFICATION OF PRODUCTION OBJECTS IN THE PROCESS OF CONTROL SYSTEMS DESIGN AUTOMATION

[Abstract of article by M. I. Abezgauz, R. I. Mayzus and A. S. Grinberg]

Authors discuss methods of building control objects based on consideration of structural properties of initial statistics. The suggested algorithm supports construction of models that consider the dynamic properties of models. Automation of construction of models is considered a process that supports selection of indicators in an automated data processing system.

8545
CSO: 1863/234

CONTROL SYSTEMS DESIGN AUTOMATION IN ORGANIZATIONS OF THE USSR MINISTRY OF HIGHER AND SECONDARY SPECIALIZED EDUCATION

Moscow AVTOMATIZATSIYA PROYEKTIROVANIYA SISTEM UPRAVLENIYA in Russian No 3, 1981 (signed to press 5 Feb 81) pp 80-87

[Article by I. M. Makarov from book "Control Systems Design Automation", issue 3 of a collection of articles, edited by V. A. Trapeznikov (chairman of the editorial board), I. V. Prangshivili, V. L. Epshteyn, A. G. Mamikonov, I. G. Dmitriyeva and D. M. Berkovich, based on papers presented at the All-Union Conference on ASU Design Automation held in Suzdal' in April 1979, Izdatel'stvo "Finansy i statistika", 12,500 copies, 208 pages]

[Text] Control systems design automation is a major national economic program for full automation of production. Considerable attention is paid to solving it in the higher school. Collective-use systems are being established in the country's VUZ's that make it possible in a new fashion to train highly qualified specialists in automatic control systems and conduct scientific research on the program for control systems design automation for the various sectors of the national economy. In the Moscow Institute of Engineering Physics, a collective-use system is being established that includes the YeS-1060, YeS-1033 and YeS-1022 computers with the corresponding peripherals (alphameric and graphic displays, plotters). At Moscow State University, development is underway on a system based on the high-throughput multiprocessor "El'brus" computer with virtual storage and corresponding peripherals.

The collective-use systems under development will allow holding classes with 400 students simultaneously in the field of automation of design of various types of complex dynamic systems. Considerable attention in the process will be paid to development of learning systems that allow instructors to give lectures and practical classes in auditoriums with extensive use of electronic digital computers.

At the MIFI [Moscow Institute of Engineering Physics], MAI [Moscow Aviation Institute imeni Sergo Ordzhonikidze] and MVTU [Moscow Higher Engineering School] imeni N. E. Bauman, standard study centers have been placed into permanent service for automated design and engineering that are equipped with alphameric and graphic displays and connected by high-frequency cables to unified series computer systems. In the study centers, laboratory exercises are carried out, homework assignments are performed and course and thesis projects are worked on; students study the characteristics of the stability, quality and precision of the control processes for various types of automatic systems (both with and without prescribed structure).

Successful operation of these study centers to a considerable extent is determined by the level of the interactive systems the designer or engineer uses to communicate with the computers. Two interactive systems have been developed at the MIFI: for editing and debugging of programs (PRIMUS) and for organization of work with application program packages under the conditions of control systems design automation (DIAL). An interactive system was developed at the MEI [Moscow Institute of Power Engineering] to analyze automatic control systems. All these systems have been implemented on YeS computers and service up to 12 displays simultaneously.

Development of software for the collective-use systems with time sharing requires organization of data banks that can be used by all the country's VUZ's. Work on establishing data banks is underway at many VUZ's. However, this large effort is still poorly coordinated and there are reasons to believe that the lack of a common technique for building a data base can lead to partial incompatibility of it. To solve this exceptionally important problem, the work of the respective VUZ's has to be coordinated centrally through the Scientific and Technical Council of the USSR Minvuz [Ministry of Higher and Secondary Specialized Education] after coordinating it with the USSR State Committee on Science and Technology.

In the country's VUZ's, this scientific research is underway: methods of automated design of systems to control aircraft for various purposes are being developed (MAI), methods of automating the processes of design and engineering of servo systems for manipulator robots are being worked out (MVTU), application program packages for analysis and synthesis of digital optimal and adaptive complex control system are being created (MIFI), application program packages for identification of control systems are being developed (MEI), and complex programs for optimization of automatic control systems are being developed (MFTI [Moscow Physico-Technical Institute]).

Much attention is being paid to creating powerful complexes for automation of processes of control systems simulation (efforts are underway at the MIFI, the MAI and the MIREA [Moscow Institute of Radio Engineering, Electronics and Automation]). In addition to the YeS computers, the complexes include analog computers. With regard to the complex nature of modern automatic control systems and the need for real-time simulation, high-speed algorithms are being developed for integrating massive systems of differential equations.

In the VUZ's, extensive training of user engineers is being conducted; they know how to perform design and engineering work by making use of systems of automated design and engineering. For successful training of cadres and performance of scientific research on automatic design systems and software suitable for them, study centers for automated design (fig. 1) are being set up on the basis of unification of the YeS computers (YeS-1022, YeS-1033 or the YeS-1035 and YeS-1060). This computer system can be developed as a result of adding on computer hardware. The first phase of the system is to be built with the YeS-1022 computer (with 512K of main storage) with the OS version that supports multiprogramming. The second phase is to include a YeS-1033 (Yes-1035) computer forming with two YeS-1022 computers the first phase of a three-machine computing complex. The third phase is to include one YeS-1060 computer, four YeS-1033 (YeS-1035) computers and eight YeS-1022 computers united into one computing system.

As seen from fig. 1, the computing system can support simultaneous operation of eight study centers. Two versions are being proposed for equipping the study centers

Russian "EC" = English "YeS"

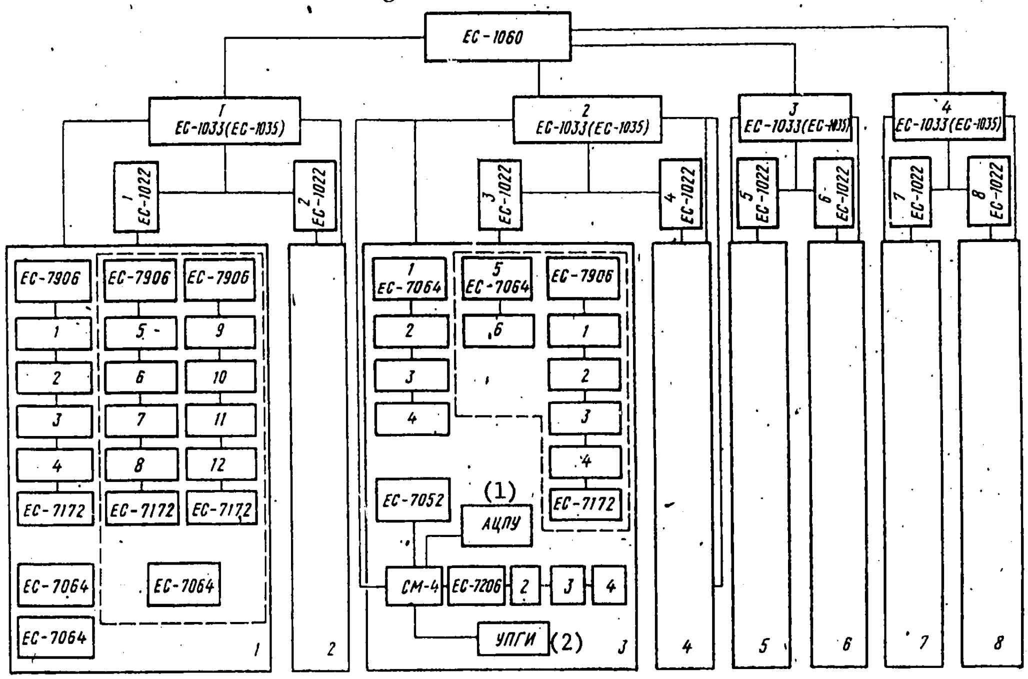


Fig. 1. Standard hardware composition for automated design study centers of the USSR Ministry of Higher and Secondary Specialized Education

Key:
1. ATsPU [alphameric printer]
2. UGPI [graphic information preparation unit]

with peripherals: one for control systems designing (frequency characteristics, transient processes, control laws synthesis) and one for control systems engineering (preparation of structural schemes and working drawings). Equipment composition is as follows:

	Study Center	
Peripherals	Design	Engineering
Alphameric Displays	12	8
Graphic Displays	2	6
Plotter	1	1
SM-4 Computer	-	1
Graphic Information Preparation Unit	-	1

In the automated design study centers, 15 students can work simultaneously; in the automated engineering study centers—13. During his period of study in a VUZ, each student must work 60 hours in the automated design and engineering study centers. Under these conditions, the proposed hardware composition will allow supporting the holding of all types of training activities for the following number of students per year:

Phase	Study Center Design	Study Center Engineering
I	500	570
II	1000	1140
III	2000	2280

Scientific research will also be performed in the study centers for automated design and engineering. When the amount of source information in the form of structural schemes and graphs is rather large, it should be input into the computer system through the SM-4 computer using the graphic information preparation unit.

The USSR Minvuz has approved the standard study center for automated design and engineering. Individual subsystems of this study center have been implemented at the MIFI, MAI and the MVTU. In 1980, all study centers in the ministry's chief VUZ's will be equipped with the standard hardware.

The problems solved in designing control systems and the composition of the software for the interactive system of automated design are illustrated in fig. 2. In the process of automated design, the following problems are solved:
linearization of dynamic equations;
control system structural synthesis (selection of composition of transducers, selection of structure of control laws, parametric synthesis);
control systems analysis (study of stability, controlability and observability, evaluation of quality and precision, as well as the effect of nonlinearity on the behavior of the control systems);
digital realization of the control laws;
mathematical modeling of the control system being designed; and
formation of output documentation (characteristics of the system and digital control computer).

Software includes the DIAL interactive system for solving problems of control systems analysis and synthesis, the interactive system for simulation and the system for preparation of output documentation.

The DIAL programming system offers the user an interactive language, facilities for creating and debugging programs written in the DIAL language, facilities for creating, storing and modifying user programs and data, facilities for operation with application program packages, and the capability of supplementing application programs with modules in the FORTRAN-IV and PL/1 languages.

It should be noted that design programs written in the DIAL language have Russian names and are stored in a special library. Thanks to this, one can express the mnemonic of the subject field in the titles of programs. In the programs themselves, modules that solve assigned problems in design can be called.

Key:

1. source data input
2. DIAL interactive system
3. linearization of dynamic equations
4. synthesis of structure of SAU [automatic control system]
5. selection of composition of transducers
6. selection of structure of control laws
7. parametric synthesis
8. system analysis
9. stability
10. controlability and observability
11. quality and precision
12. consideration of nonlinear characteristics
13. analysis of self-excited oscillations
14. digital realization of control laws
15. decision
16. repeat calculation
17. complete calculation
18. interactive simulation system
19. mathematical modeling
20. refine calculation
21. documentation system
22. documentation
23. characteristics of SAU [automatic control system]
24. characteristics of control computer

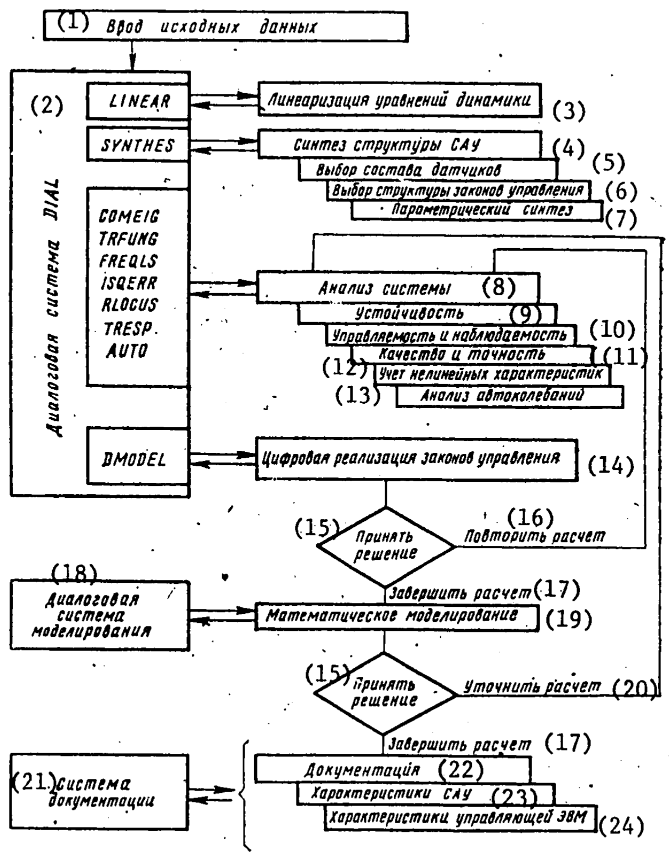


Fig. 2. Control systems design automation software

The main modules are:
linearization of equations by the method of least squares;
synthesis of control laws by the quadratic criterion of quality;
computation of eigenvalues and vectors of matrix of system under study;
computation of transfer function;
computation of frequency characteristics;
computation of integral of square of error;
computation and construction of root locus;
computation and construction of transient processes;
analysis of self-excited oscillations; and
computation of discrete laws of control.

The interactive system services simultaneously up to 16 users at remote terminals. The DIAL design system operates in the mode of interpretation, i.e. each statement of this program is stored in source form and executed immediately. Thanks to this, the traditional stages of the batch mode of processing are eliminated (compilation and editing) which makes it possible to quickly solve engineering problems and in a short time debug the needed programs to design control systems.

As a result of automated design of the control system with any structural scheme (described by differential or difference equations up to the 100th order), optimal laws of control can be found, and logarithmic frequency characteristics, amplitude-phase characteristics, root loci and transient processes can be built. The corresponding characteristics derived as output documents on the graphic plotter are shown in fig. 3. From the figure it follows that 40 seconds of machine time is required to construct logarithmic amplitude and phase frequency characteristics for 10 points of amplitude-phase characteristics, as well as characteristics of a transient process, and 90 seconds of machine time is needed for building the root locus. Consequently, automation of the process of control systems design is performed about 20-fold faster than with conventional methods. The precision and validity of results derived are also enhanced.

Other automated design systems are under development too in institutes of the USSR Minvuz. Thus, for example, at the MEI, the MASS software has been developed for automated modeling of continuous systems of automatic control described by small orders of differential equations or specified structural schemes. The MASS system software uses a language of directives and statements which makes it possible to put into the computer the structural scheme of the system under study, to effect input of parameters of this system, to achieve conversion of the input structure, to perform integration of differential equations and perform parametric optimization. Program results are tables and graphic materials output on an alphameric printer.

Language directives are used to organize user communication with the MASS system and control of sequence of data input and computation process. Statements are intended for input of source information. The MASS program system has been implemented for M-6000 computers and YeS computers with the disk operating system.

Software developed at the Chelyabinsk Polytechnical Institute makes it possible to carry out calculation of continuous discrete as well as nonlinear systems of automatic control with YeS computers also by using the language directives. The distinguishing feature of the program complex is the capability of making a working design of electrical machines for actuating mechanisms of automatic control systems.

At the MVTU imeni Bauman, efforts are underway to automate servo systems by using application program packages for modeling and calculation of systems in the batch processing mode. At the Leningrad Institute of Precision Mechanics and Optics, software has been developed for designing optical-electronic instruments and devices of control systems.

The USSR Minvuz coordinates efforts of the VUZ's in the field of development of automated systems for design and engineering. For this purpose, under the NTS [Scientific and Technical Council] of the ministry, the Scientific Methodological Council on SAPR [Automated Design Systems] has been established; this council has prepared the necessary technical documentation on systems for automated design and engineering of control systems and has introduced it in a number of VUZ's and scientific research institutes of the industrial ministries.

A complex program of efforts has been worked out to develop automated teaching systems which will allow raising the quality of instruction of students in the field of automated control systems. Data banks will be established in these systems that the VUZ's can exchange with each other, since subsequently it is planned to combine the computer centers of our own VUZ's into a common computer network.

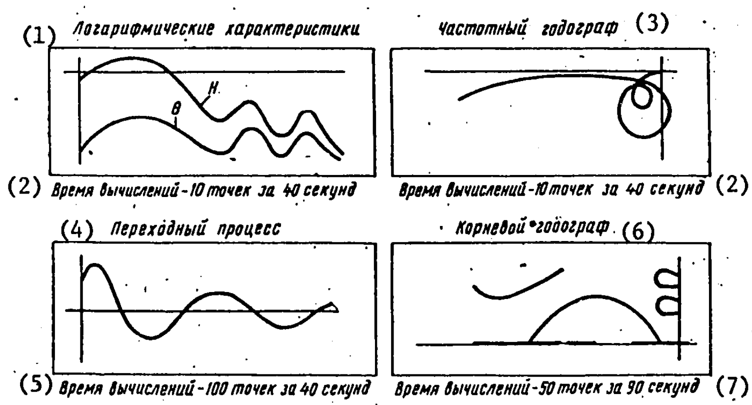


Fig. 3. Output characteristics of control systems being designed (function is analysis of dynamic systems; number of programs is 12, package size is 160K)

Key:

1. Logarithmic characteristics
2. Time of calculations for 10 points is within 40 seconds
3. Frequency locus
4. Transient process
5. Time of calculations for 100 points is within 40 seconds
6. Root locus
7. Time of calculations for 50 points is within 90 seconds

8545
CSO: 1863/234

SOME EVALUATIONS OF EFFICIENCY OF PARALLEL COMPUTATIONS

Moscow AVTOMATIZATSIYA PROYEKTIROVANIYA SISTEM UPRAVLENIYA in Russian No 3, 1981 (signed to press 5 Feb 81) pp 88-100

[Article by E. A. Trakhtengerts from book "Control Systems Design Automation", issue 3 of a collection of articles, edited by V. A. Trapeznikov (chairman of the editorial board), I. V. Prangshivili, V. L. Epshteyn, A. G. Mamikonov, I. G. Dmitriyeva and D. M. Berkovich, based on papers presented at the All-Union Conference on ASU Design Automation held in Suzdal' in April 1979, Izdatel'stvo "Finansy i statistika", 12500 copies, 208 pages]

[Excerpt] In conclusion, let us note that with parallel computations, a substantial increase in computation rates can be achieved by converting in the translation process cyclic computations of scalars into vector instructions. At present, several algorithms for such conversions have been proposed [21], their convergence has been proven and they have been implemented in translators. These algorithms are complex and do not convert all types of loops.

In computing systems with multiple flow of instructions, several sections of programs can be executed simultaneously. As a function of the structure of computer systems, the interrelation of flows of instructions may vary since each flow of instructions is a completely independent problem. Flows of instructions can form subtasks of one problem. In the latter case, the association by data is preserved between subtasks, but each subtask is, as a rule, a rather large program and is poorly associated with the other subtasks by control.

Finally, a program may be subdivided into relatively small parts, sufficiently closely associated with each other by both data and control. Subdivision of programs into these concurrently executable sections can be performed directly by the programmer or automatically by the translator.. An algorithm for formal subdivision of programs into these sections is described in [22].

The emergence of concurrently executable sections of programs requires development of special mechanisms to synchronize the computing process. This problem arose already in the stage of multiprogramming and to solve it there is a rather well developed apparatus of semaphores and monitors that accomplish the task [23-27]. These facilities are relatively easily implemented in translators, and a large number of linguistic constructions has been developed to describe them [28].

8545
CSO: 1863/234

- END -