JPRS L/9703

30 April 1981

USSR Report

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 13/81)

FBIS FOREIGN BROADCAST INFORMATION SERVICE

FOR OFFICIAL USE ONLY

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300100066-5

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.

APPROVED FOR RELEASE: 2007/02/08: CIA-RDP82-00850R000300100066-5

JPRS L/9703

30 April 1981

USSR REPORT

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

(FOUO 13/81)

CONTENTS

PUBLICATIONS

GENERAL

LETTER TO GENERAL SECRETARY OF CPSU CENTRAL COMMITTEE, CHAIRMAN OF PRESIDIUM OF USSR SUPREME SOVIET, COMRADE LEONID IL'ICH BREZHNEV

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 p 2

[Letter to general secretary of CPSU Central Committee, chariman of Presidium of USSR Supreme Soviet, Comrade Leonid Il'ich Brezhnev from Ministry of Instrument Making, Means of Automation and Control Systems]

[Text] Dear Leonid Il'ich! We are glad to report to you that, fulfilling the decisions of the 25th CPSU Congress and your instructions on economic problems and being guided by the decree of the CPSU Central Committee "On the socialist competition to meet the 26th CPSU Congress with honor," the workers, scientists, engineering and technical personnel and employees of the Ministry of Instrument Making, Means of Automation and Control Systems completed fulfillment of the plan of the 10th Five-Year Plan ahead of schedule by rates of growth in the volume of production. The volume of production of instruments, equipment, means of automation, computer equipment and other products of instrument building increased 1.7-fold during the five-year plan. Labor productivity during the five-year plan as a whole will increase by more than 52 percent, which exceeds the task of the five-year plan. The cost of produced products will decrease by more than 10 percent. The tasks of the five-year plan for production of the most important nomenclature and national consumer goods are also being fulfilled ahead of schedule.

The high work indicators of the sector are the result of extensive use of the advances of scientific and technical progress, development and introduction of modern equipment into production, of leading methods of labor organization and progressive technology and of improving the economic mechanism and economic methods of management. More than 2,500 of the most important types of instruments, means of automation, computer equipment and national consumer goods were produced during the five-year plan; more than 1,500 automated control systems, including 350 ASU [Automated control system] for production processes in the leading sectors of the national economy, were put into operation. The volume of output of articles using microelectronic equipment--integrated circuits, microprocessors and microcomputers --increased more than threefold and exceeded two billion rubles in 1980. In this case the main operating characteristics of the instruments and facilities of computer equipment--accuracy, speed and reliability--were improved significantly. Of the total volume of products produced, approximately half comprise articles assimilated during the 10th Five-Year Plan. A program is being realized to organize and develop production of numerical program control devices for metalworking equipment and robot manipulators and instruments for scientific research, agriculture and environmental monitoring.

Approximately 50 percent of the collectives of sector enterprises reported on fulfilling the tasks of the 10th Five-Year Plan ahead of schedule. Among them are the Moscow PO [Production Association] Manometr, the PO Leningrad Electrical Engineering Plant, the Kiev PO Kochelectropribor, the Orel PO Prompribor, the Moscow NPO [Scientific Production Association] Spektr, the Kishinev NPO Valna, the Minsk Clock Plant imeni 60-letiya of the Belorussian Communist Party and others.

The workers, scientists, engineering and technical personnel and employees of the sector, realizing the tasks advanced by the party at the October (1980) Plenary Session of the CPSU Central Committee, are laboring to fulfill the new increased socialist pledges in honor of the 26th CPSU Congress. Instruments, equipment, means of automation and computer equipment worth no less than 310 million rubles and national consumer goods worth 117 million rubles will be produced above the five-year task prior to the end of the current year. A number of investigations important to the national economy will be fulfilled ahead of schedule by the opening of the Party Congress including the fact that state trials will be completed and industrial production of highly productive computers for geophysical computer complexes for processing the data of geological oil and gas prospecting work will be begun. National consumer goods worth no less than 10 million rubles will be manufactured above the plan for two months of 1981.

The instrument builders assure the CPSU Central Committee and you personally, Leonid Il'ich, that they will struggle persistently to increase production efficiency and work quality, to fulfill the plans of the party in building of communism and will welcome the 26th Party Congress with honor. (27 November 1980)

6521
CSO: 1863

INSTRUMENT BUILDING FOR THE 26th CPSU CONGRESS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 2-5

[Article by Doctor of Technical Sciences M. S. Shkabardnya, USSR minister of instrument building, means of automation and control systems]

[Text] Implementation of the course planned by the 25th CPSU Congress to increase production efficiency and product quality and acceleration of scientific and technical progress are related to harmonious development of the entire national economy of our country. Reducing the fraction of manual labor, support of complex mechanization and automation and improving methods and means of management are conditions for increasing material production efficiency. Solution of these most important problems posed by the party makes no sense without modern development of instrument building, computer technology and electronics. It is the given complex of sectors on the eve of the 10th Five-Year Plan that was identified by Comrade L. I. Brezhnev as the "catalyst of technical progress" and as the tasks of the five-year plan that provided for their leading development.

The instrument builders successfully completed fulfillment of the tasks of the 10th Five-Year Plan ahead of schedule. The selfless labor of our workers, engineers and scientists brings visible fruits. The extensive qualitative progressive changes in the structure of production, technical level and organization of the instrument building industry are the results of this labor.

Instrument building during the 10th Five-Year Plan compared to the previous period was developed more intensively and is characterized by a significant expansion of the nomenclature of instruments and means of automation according to the growth of the needs of the national economy, especially such leading sectors as power engineering, chemistry, nonferrous and ferrous metallurgy, machine building and so on, according to an increase of their technical level, accuracy and reliability, to an expansion of functional capabilities and to an increase of production volume.

The results of fulfilling the state plan and socialist pledges for 1976-1980 indicate that the collectives of the enterprises and organizations of Minpribor [Ministry of Instrument Making, Means of Automation and Control Systems of the USSR] coped successfully with the tasks posed to them.

The growth of production volume was 171.3 percent compared to 167.1 percent provided by the five-year plan, with an average annual growth rate of 11.35 percent (10.8 percent according to the plan) and labor productivity increased 1.52 times. More

than 70 percent of the growth of product output was achieved as the result of an increase of labor productivity. More than 2,500 new articles were developed and assimilated and more than 1,500 obsolete articles were taken out of production. Approximately 40 percent of the total product volume of the sector was produced with the state Emblem of Quality. As in previous years, production of means of control computer equipment, output of which increased 1.8-fold in 1980 compared to 1975, developed at the fastest rates. The production of instruments for monitoring and regulation of production processes and also of electric measuring instruments increased 1.8-fold, that of instruments for measuring mechanical values and weight-measuring and weight-proportioning devices increased 1.7-fold and that of instruments for scientific research increased 1.75-fold.

Much has also been done by instrument builders in developing the production of consumer goods: clocks, fountain pens, typewriters, jewelry, games, stationery and writing implements. The sector is now manufacturing several thousand types of these articles and approximately 80 percent of them have been renewed or newly assimilated, while the volume of their output increased 1.5-fold during the five-year plan.

Based on the systems approach, formulated during previous years, to establishment and development of production and complex use of devices and means of automation of different designation in the national economy, the technical policy of the sector oriented toward increasing the efficiency and expansion of utilization of modern methods and means of measurement, monitoring, automatic control and information processing, is being formulated and implemented in the form of a number of large purposeful programs.

They include a complex program of work to establish and organize production of an international small computer system--SM EVM, carried out by enterprises of Minpribor jointly with a number of enterprises of the socialist countries. More than 200 new devices of this unit system of computer equipment with developed peripheral equipment and software has been developed and assimilated. The enterprises of the sector have organized large-serial production of control computer complexes (UVK) of the new generation of SM EVM designed for automated production control systems, scientific experiment management systems and automated design control systems. At the same time, the laboriousness of producing most models of the SM EVM has been reduced more than one-half compared to previously produced machines, which made it possible to increase significantly the output of UVK of SM EVM in 1980. The serial output in various configurations and in the form of problem-oriented computer complexes creates a strong scientific and technical base for extensive use of computer equipment in this class for automation of production and labor processes of the most diverse nature in all sectors of the national economy.

An extensive program of work is being implemented to develop new seismic prospecting equipment and specialized computer complexes for processing geophysical information, the use of which will make it possible to increase significantly the efficiency of geological prospecting for oil and gas and also to evaluate and analyze the status of natural resources on large scales. New field digital seismic prospecting stations Progress have been developed and put into production and multiprocessor geophysical expedition computer complexes PS-2000 and other equipment have been developed within the framework of this program.

The production of devices and systems for numerical program control (ChPU) of metal-cutting machines and industrial manipulator robots, including those on a modern

microelectronic component base with built-in microprocessors and microcomputers, is being developed at high rates. Production of ChPU devices of different designation increased 2.6-fold during the 10th Five-Year Plan and models of them were completely renovated. More than 65 percent of the ChPU devices are being produced with the state Emblem of Quality. The decree of the CPSU Central Committee and the USSR Council of Ministers adopted at the beginning of 1980 "On a significant increase of the technical level and competitiveness of metal working, casting and woodworking equipment and tools" also provided for accelerated development and production of ChPU devices for all main groups of industrial machining equipment (machine tools of various types, forging-press and foundry equipment, automatic lines and machining centers and sections) for subsequent years. The existing scientific basis in machine control theory and in production operation programming theory and the extensive experience of development and complex debugging of complex electromechanical machine tool systems--UChPU--permit complex solution of these problems at a high scientific and technical level.

The development and further improvement of the State System of Industrial Devices and Means of Automation (GSP), which encompasses all the most important functional groups of measurement, monitoring and regulation hardware for production processes and which comprise the basis for construction of ASU TP [Automated production process control system] of various classes and designations, was continued during the 10th Five-Year Plan. Based on GSP standards and using new physics and technological principles, a large number of types of sensors of physical values (pressure, flow rate, level, temperature, electric output and so on), normalizing and functional measuring converters, servo mechanisms, local and group regulators, means of telemechanics and other devices, including those oriented toward use under severe operating conditions, was developed and put into production.

Expansion of industrial output of efficiently built series of GSP instruments and devices technically compatible to each other made it possible to design and use the most diverse and complex monitoring, regulation and control systems for the needs of practically all sectors of industry with analog and analog-digital nature of production and primarily of power engineering, ferrous and nonferrous metallurgy, chemistry, oil and gas production and petroleum refining.

The nomenclature of GSP devices, considerably renewed during the 10th Five-Year Plan, numbers more than 2,000 types of instruments and devices and their production volume reached approximately two billion rubles in 1980. The main specifications--precision of monitoring and regulation, ranges of measured values and reliability--improved significantly. GSP devices and specialized unitized complexes of instruments built with observation of GSP standards now make it possible to measure and regulate more than 600 varieties of physical values and production parameters (including those by indirect methods) over a wide range of values and under different operating conditions.

The domestic GSP, coordinated by basic principles and standards with the national systems of CEMA members, has no equal in the worldwide practice of instrument building in functional completeness, scope of measurement, monitoring and regulation problems, universality of applications, level of unitization, unification and standardization, metrological support and scales of production and use in the national economy.

An extensive program of investigations to establish and organize output of measuring devices and means of automation for different industries in agriculture and processing of agricultural products has been organized and is being implemented successfully. The output of complete agrochemical laboratories, automated weighing and weighing-proportioning equipment, analytical equipment, means of automation for animal husbandry complexes and for irrigation systems and other measuring and regulating devices has been expanded.

The nomenclature of devices supplied to agriculture in 1980 numbered more than 350 and investigations are continuing in this field.

One of the most important statewide programs consists of a complex of investigations to establish and develop devices for scientific research and means of automation of scientific experiments.

Realization of the most important national economic task posed by our party on transforming science into a direct productive force, increasing the rates of development and intensifying the effect of achievements of the fundamental sciences on the rates of technical progress in the national economy required accelerated development from instrument builders in developments and assimilation of production of many of the most complex devices and equipment for outfitting scientific institutions.

An extensive program of investigations to establish and organize production of many types of devices for scientific research (including those based on new physical principles), new means of automation of scientific experiment using mini- and microcomputers and programmable digital equipment for measurement and analysis of the composition and structure of substances and materials, was prepared and is being successfully implemented jointly with the USSR Academy of Sciences. A considerable number of scientific organizations and enterprises of the sector participates in development and production of devices for scientific research. More than 100 types of complex devices for scientific research, including mass and radio spectrometers, x-ray analyzers, chromatographs, electrometers, electron microscopes, structural analyzers and other scientific apparatus, were developed and put into production during the 10th Five-Year Plan jointly with institutes of the USSR Academy of Sciences.

Means of automation of scientific experiment--measuring and calculating complexes with programmable measuring apparatus in CAMAC standards and developed software and metrological support--have been established and are being produced. More than 42 percent of the devices for scientific research are produced with the state Emblem of Quality. According to the draft of the "Main trends of economic and social development of the USSR for 1981-1985 and for the period up to 1990" and according to the growing needs of Soviet science, this complex of scientific research and experimental design work, which enriches our design organizations with new progressive ideas and which actively affect an increase of the technical level of all instrument building, will be developed at high rates during the forthcoming five-year plan.

A great deal of attention was devoted in the sector during the 10th Five-Year Plan to further development of production and increase of the technical level of the clock industry--the traditional basic sector of precision mechanics. Production of

service clocks was increased 1.2-fold during the five-year plan and reached 58 million units in 1980. Models of mechanical clocks produced were completely renovated and production of highly accurate electronic crystal watches was organized. Much has been done to improve the technology of clock production: automated assembly of clocks using manipulator robots (USSR State Prize for 1979), laser machining of jewels, automatic timing of clocks using computers and much more were organized. However, even greater changes in this traditional field are expected during the 11th Five-Year Plan with regard to a further significant growth of the specific weight of electronic clocks and the entry of the modern advances of electronics into the clock industry.

An important factor in development of the sector and its active effect on the rates of technical progress in the national economy was further expansion of work in the field of development and introduction of automated control systems (ASU). ASU were developed for all levels of control: production units, plants, enterprises, combines and production associations, sectors, nonindustrial facilities and territorial complexes. The range of problems solved was expanded significantly and the technical level of ASU for production processes and sector ASU was increased. Systems based on modern scientific principles of construction and functioning were established: direct digital control, adaptive and integrated, multilevel hierarchical and also complex queueing systems and the first "distributed control" systems. Such large ASU as the ASU of the PO [Production Association] Uralmash, PO Rostsel'mash and the complex of automated systems of ASU-Olypiada became operational.

Automation of control of production processes, units and plants based on the use of modern computers is the main trend of scientific and technical progress in sectors of industry with analog and analog-digital nature of production. The accumulated experience clearly shows that development of ASU TP for many large-capacity production facilities now operating ensures a significant increase of their operating efficiency by operational optimization of control processes, which is essentially impossible when using traditional methods of monitoring and local automation.

The scientific research and planning organizations of Minpribor concentrated their efforts during the 10th Five-Year Plan primarily on development of pilot ASU for highly productive large-capacity units of new types in power engineering, ferrous and nonferrous metallurgy and the chemical, petroleum refining and petrochemical, the petroleum, gas, coal, pulp and paper and construction materials industries. More than 150 highly efficient pilot ASU TP for new production facilities in the indicated sectors were established and put into operation through the efforts of our scientific collectives jointly with the customers.

Among them are ASU TP of nuclear energy blocks with capacity of 440, 600 and 1,000 MW at the Beloyarskaya and Novovoronezhskaya Nuclear Power Plants, the ASU of the world's largest thermal energy unit, rated at 1,200 MW, at the Kostromskaya GRES, the largest blast furnace No 9 of the Krivorozhstal' Plant imeni V. I. Lenin, the universal beam rolling mill of the Nizhniy Tagil' Metallurgical Combine imeni V. I. Lenin, the concentration plants of the Zyryanovsk Lead Combine, gas lift production of petroleum at the Association Kaspmorneft' and many others.

Domestic and foreign experience shows that the greatest deficiency in utilization of all modern methods of control, monitoring and regulation of the corresponding hardware--from control computers to servo mechanisms--is achieved in the national economy with complex, systematic organization of interaction of this hardware with the

monitoring and control object, including the technological personnel of this facility. Therefore, the activity of our scientists, designers, planners, installers and developers of devices and means of automation who participate in establishment, introduction and subsequent development of ASU of various classes and designations, is the most important essential aspect of the activity of the sector. Significant results were achieved during the 10th Five-Year Plan in improving the methods of this work.

Based on generalization of the developments of a large number of basic objects, standard design solutions and applied program packs used extensively by organizations and enterprises of other sectors in development of systems, were developed. Methods of ASU design and and software for them, scientific methods of building integrated and multilevel ASU that encompass organizational-production and technological processes at enterprises were worked out and developed. All-Union stock of general-purpose applied program packs and libraries of standard algorithmic and program modules for ASU TP of a number of sectors of industry were established and developed, automated software generation systems for ASU TP were put into operation and scientific experimental centers for testing pilot systems are functioning. Further development of the indicated trends and a significant increase of the finite national economic effectiveness in complex utilization of all means and methods of modern automation are one of the most important tasks of the instrument builders during the 11th Five-Year Plan.

Fulfilling the decisions of the 25th CPSU Congress, the instrument builders devoted special attention to development and implementation of measures which enhance the technical and economic efficiency of production. Specifically, they include further improvement of management of the sector as a whole and of its individual enterprises, an increase of the level of the operational rhythm of plants, introduction of highly productive automatic equipment and new production processes, introduction of the complex product quality control system at practically all plants, improvement in the use of basic funds and capital investments, extensive use of microelectronics in development of new devices and means of automation, including organization of certain types of integrated circuit production, development and introduction of ASU TP at plants of the sector, development of the instrument base of the sector, local production of nonstandardized production equipment and specialized cooperative production of assemblies and parts for general sector application (casting, fastenings, sockets, construction materials, printed circuit cards and so on). Modern methods of automated design and monitoring (using computers) of printed circuit cards and installation panels have been developed and are being used successfully at many plants, design of production equipment is being automated at some plants and machine output of production documentation is being organized.

Implementation of measures to increase production efficiency made it possible to complete fulfillment of the plan of the 10th Five-Year Plan on the growth of product volume ahead of schedule. Devices, means of automation and computer equipment worth more than 310 million rubles were produced above the five-year plan. The task of the five-year plan on increasing labor productivity was overfulfilled. Consumer goods worth 117 million rubles above the task for the five-year plan were manufactured. A total of 53,000 tons of ferrous rolled metals, 40,000 tons of nonferrous rolled metals, 57.3 million kW·hr of electric power and 100,000 Gcal of thermal energy were conserved during the 10th Five-Year Plan. The scientific research institutes, design offices and plants of the sector fulfilled 283 tasks of the state

plan for development of the national economy of the USSR on assimilating the production of the most important devices and means of automation of new types during the five-year plan, including 55 ahead of schedule, and more than 100 tasks on 89 problems according to programs of GKNT [State Committee for Science and Technology] on solution of the most important scientific and technical problems.

Considerable work was carried out on automation and mechanization of production at plants of the sector: 80 shops were mechanized in a complex manner, more than 650 mechanized continuous production conveyor lines were put into operation, 250 automatic and semiautomatic lines and more than 450 machine tools with ChPU were put into operation.

The decree of the CPSU Central Committee "On the socialist competition for a worthy welcome of the 26th CPSU Congress" aroused new creative enthusiasm among the collectives of instrument builders, who organized a socialist competition to fulfill the tasks of the 10th Five-Year Plan ahead of schedule and new labor achievements. The brigades, sections, shops, enterprises and organizations adopted increased socialist pledges and strived to work in a shock-labor and communist manner.

Among the collectives in the vanguard of instrument builders are the Second Moscow Clock Plant, the Kiev Production Association Tochelektropribor and Production Association Elektronmash imeni V. I. Lenin, the Moscow Production Association Manometr, the Orel Production Association Prompribor and the Production Association Leningrad Electromechanical Plant. These and many other collectives reported fulfillment of the tasks of the 10th Five-Year Plan ahead of schedule.

Thus, the collective of the Kiev Production Association Tochelektropribor completed the tasks of the 10th Five-Year Plan in labor productivity in July 1979 and the tasks in production volume on 3 October 1980. They produced and sold products worth more than 15 million rubles above the tasks of the five-year plan. The entire increase in the volume of production was achieved as a result of increasing labor productivity.

Many thousand workers of the sector completed their personal five-year plans ahead of schedule. More than 35,000 leading workers, approximately 400 brigades, 100 sections and shops reported fulfillment of their own pledges to complete the tasks of the 10th Five-Year Plan as early as the 110th anniversary of V. I. Lenin's birth. More than 80,000 production leaders successfully completed the 10th Five-Year Plan by the 63rd anniversary of the Great October Socialist Revolution.

Products worth 85 million rubles above the plan were sold throughout the sector as a whole in 1980, which is 25 million rubles more than was envisioned by the additional pledges adopted in honor of the 26th CPSU Congress.

The scientific research, design-production and planning organizations of the sector have also made a significant contribution to the increase of production efficiency during the socialist competition in honor of the 26th CPSU Congress. The scientists, engineers and workers of the Institute of Control Problems (Moscow), the Severodonetsk NPO [Scientific Production Association] imeni 25th CPSU Congress, NIIteploprибor [Scientific Research Institute of Heat Power Engineering Equipment], VNII [All-Union Scientific Research Institute] of Electric Measuring Devices (Leningrad), the Vilnius Production Association Sigma, MNIPI [Moscow Scientific Research and

Planning Institute of Network Program Planning and Control Systems for Industry and other leading organizations of the sector came to the congress with important achievements in development and assimilation of new equipment at the level of the best worldwide models.

Being guided by the draft of the "Main trends of economic and social development of the USSR for 1981-1985 and for the period up to 1990," published by the CPSU Central Committee, the instrument builders aim for new boundaries in development of the sector with high labor enthusiasm.

Intensive development and a further increase of production of instruments, means of automation, high-speed control and computer complexes, primary development of progressive and more modern types of instrument production and achievement of higher qualitative level and efficiency of ASU under development are planned. Extensive automation and complex mechanization of instrument building processes will be implemented and the fraction of manual labor will be reduced significantly, including that on specific, "thin" assembly and monitoring-checking operations. Microelectronic technology and modern microprocessor equipment will become the determining base for development of instrument building in all its main trends.

The new level of technical equipping not only of sectors of heavy industry and machine building but also of agriculture, the construction industry, transport, medicine, public services and environmental monitoring during the 11th Five-Year Plan require development and production of many types of essentially new devices and means of automation oriented toward the specifics of each user. This is primarily related to sensors of physical values and different production parameters, which largely determine the capabilities of developing highly efficient modern control systems. Special attention will be devoted to development and production and to significant expansion of the efficient nomenclature of various types of sensors.

The most important integral characteristic of the technical level and quality of instrument building products is the operational reliability of the devices and means of automation. The significant results achieved in this direction permit us to plan new tasks for the 11th Five-Year Plan for all groups of articles with regard to finite national economic estimates of this activity, which encompasses all stages of development, production and maintenance of means of automation.

Welcoming the 26th CPSU Congress with high labor enthusiasm, the instrument builders assure the CPSU Central Committee that they will make a worthy contribution to implementation of the program planned by the Party for further development of the national economy of our socialist motherland.

6521
CSO: 1863

UDC 65

SOME PROBLEMS OF SCIENCE IN DEVELOPMENT OF CONTROL SYSTEMS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 6-8

[Article by Academician V. A. Trapeznikov, director of Institute of Control Problems]

[Text] The science of management, which occurred approximately 100 years ago, began to develop especially rapidly during the 1950s and now encompasses a wide range of problems related to management of engineering, socioeconomic and biological objects.* The science of management encompassed a number of new fields during the past few years. These are problems of classical sections of the theory and those sections which one may also now regard as classical: optimum, adaptive and logic management, the problem of stochastic systems, identification, machine modelling, construction of man-machine complexes, design automation and automation of scientific research. Management theory has penetrated many fields of the industrial and nonindustrial sphere of the national economy and has encompassed management hardware, especially that developed on the basis of computers. Such achievements as missile and spacecraft control, automatic docking and solution of many other related problems became possible only on the basis of automatic control theory.

Let us dwell on some of the future problems of development of science and technology.

As is known, science is developing in an evolutionary manner, step by step, but jumps occur from time to time which introduce a new quality into the process of development. I would like to dwell on these intermittent points of its growth. On the other hand, without falling into predictions of fantasy, one can talk about "intermittent" trends that already have adequate scientific and technical bases that are real for use in the national economy during the 11th Five-Year Plan.

It is known that there were several steps of scientific and technical transformations related in one way or another to management problems in the development of mankind. The first industrial revolution was characterized by replacement of physical human labor by the work of machines. It subsequently turned out that many machines, especially steam machines, are unable to operate without constant control and this caused the appearance of automatic regulating devices which freed man to a significant degree from performing a number of functions of mental activity. Automatic control systems gradually encompassed various fields. Control hardware was also improved simultaneously, freeing the mind of man. Finally, computers appeared which provide an enormous jump, specifically, in the sphere of control.

---

* Based on the author's report at the Eighth All-Union Conference on Control Problems at Tallinn, held on 6 October 1980.

We feel that a period has now come when computers and other devices should relieve man of part of his mental activity in development of control systems themselves, which is undoubtedly a qualitative jump. The appearance of new hardware and the fundamental change of the existing structures or configurations of control systems usually result in the appearance of new theoretical approaches and this in total provides a jump in development of our science. These problems will be primarily touched on in this article.

Let us formulate some requirements on control systems which reflect the interests of the national economy and on the other hand, which take into account real situations encountered in life. These are primarily the characteristics of the producer and user.

From the national economic viewpoint, there are no principal antagonistic contradictions in our social system between the user and the producer, since they are links of a common closed sphere of circulation of property, energy and information and they work for the common good of the state. However, two psychologically different approaches to the concepts "consciousness," "discipline" and so on can be noted in real life, and namely: the approach of the producer and the approach of the user. The producer may be a plant producing a means of control and KB [Design office] and NII [Scientific research institute] which produce designs, algorithms and programs.

Unfortunately, the following principle is predominant in the producer's psychology in many cases: fulfill the plan according to the confirmed technical tasks, turn over the product, perhaps even with reduced quality, and sign the report on fulfillment of the plan. What will happen to the product in the future and who will correct the deficiencies is of little concern to the producer in many cases. Let us assume that almost everyone has encountered a similar situation, specifically upon introduction of our ASU [Automated control system], without mentioning construction facilities.

With regard to the specifications for a product, they are usually closer to the concept "what can be done" than to the concept "what needs to be done."

The user's position is diametrically opposite. He wants to have a product that is at a modern level so that operation of the product proceed with the least fuss. Exactly one of the reasons that make it difficult to utilize control systems is failure to consider the user's requirements. An attempt to force the user to use systems inconvenient for him by administrative pressure or planning measures is unsuccessful and only creates a vision of activity.

In the final analysis the producer's and user's positions reduce to problems of quantity on the one hand and quality on the other and to contradictions related to both concepts. The fact is that a high-quality article will always be more labor-consuming than a low-quality article at some level of design, technology and organization of production.

I am in the position of the user and I will formulate the general requirements which the user places on control systems from this viewpoint.

The life of a product related to control usually consists of three phases: 1) development, which may include design and adjustment, 2) operation and 3) modernization related either to a change of the controlled medium or to the appearance of new, more improved solutions which sometimes appear even during design. The user's requirements to which I turn are related to all three phases of a product's life.

We feel that a product in general use, without mentioning its quality, should meet at least four criteria.

The first criterion is simplicity and convenience of use. This means that there should be the capability of assimilation without recruitment of scarce, highly qualified specialists. The periods of training and assimilation should be minimal, preferably on the order of several weeks. Of course, this does not contradict the need to increase the general scientific level of specialists.

The second criterion is the capability of modernization. It is known that the structure of a controlled object, its properties and finally the volumes and formats of the accompanying documentation are subject to various changes during operation and sometimes even during design. If a control system does not provide for these capabilities, it passes from a progressive to a conservative factor.

The third criterion is viability. This concept is somewhat broader than that of "reliability," since it includes a number of concepts of "reliability," including the concept of the probability of failure-free operation. The requirements on the average length of failure-free operation depend on the controlled object, which should be calculated in thousands of hours under ordinary conditions, with the capability of rapid diagnosis of the point of damage and rapid replacement of the damaged assembly. The total lifetime of a control system in industry should be determined at least as 10-15 years. The concept "viability" envisions the efficiency of a system, perhaps with some deterioration of control quality with external emergency effects on the system, while the theory of viability specifically considers the dynamics of the development of emergencies.

The complexity of control systems increases as the complexity of controlled objects increases. For example, the control systems of nuclear power plants may contain thousands and sometimes tens of thousands of sensors and other similar components. It is clear that a large number of components leads to a reduction of the viability of a system. Science noted the ways of increasing viability by redundancy, voting, diagnosis, correct design of the structure with regard to more or less important sections and finally by finding the most reliable methods of measurement.

However, science alone cannot provide the proper reliability of complex systems. The sources of reliability lie in correct design, use of high-quality materials, good technology and organization of operation by operation checking during manufacture. This makes a control system more expensive and increases the laboriousness of its manufacture, but this increase in cost will be repaid with interest by reducing the idle times of the basic expensive controlled equipment, for example, an electric power plant. One can say that "quality is more economical than quantity" from the state's viewpoint.

We note that the proper significance is frequently not given to problems of reliability and viability. Thus, for example, one can point out cases when diagnostic

and some other circuits were thrown out for purposes of simplification in development of computers. It is clear that this led to a sharp reduction of the operational characteristics of the machine.

The fourth criterion is economy provided by low capital investments, low operating expenses and gradual introduction of the control system into operation, beginning with the most efficient tasks. When calculating economy, one should not forget the different times of initial expenditures and the economy achieved.

It is known that the relative cost of hardware in a control system is reduced each year, while the fraction of the cost of design and programming increases, reaching 60-70 percent of the total expenditures for the system. Some foreign companies even offer computers to users free of charge, requiring payment only for the software. Development of automated design, programming and debugging systems is becoming exceptionally timely. It is for this reason, talking about the possible points of growth, that one should dwell on these problems, i.e., on problems which assist the intellectual creative activity of man.

The process of design is diverse and strictly speaking begins with research and predesign work in laboratories or under other conditions. A qualified design requires preliminary analysis of the facility to be automated, gathering of its characteristics and compilation of a model. This is a laborious and prolonged process and identification devices connected to a production or different object without interruption of its functioning, which may relate to automation of the experiment, come to one's aid here.

Turning to automation of design as to one of the important points of growth, its diversity should be noted. Automation of design begins with design of components such as microcircuits and BIS [Large-scale integrated microcircuit], design of machine assemblies and finally ends with design of entire systems, for example, of control systems.

The concept of SAPR [Automated design system] is rather diffuse and usually includes design of material systems: microcircuits, tools, machine tools, architectural structures and so on. During the past few years SAPR, utilizing machine graphics in most cases, have been developed throughout the world. They have been developed by tens of NII and KB in our country. Unfortunately, the exchange of information in the field of SAPR is difficult since some leading specialists regard it as a professional secret. As a result, there is no unanimity and standardization of requirements on SAPR, they are developed with respect to various types of computers and different classes of problems, have different degree of complexity and sometimes require prolonged assimilation.

As a prospect, one should turn attention to development of machine graphics systems which provide effective interaction and dialogue of the user with the computer, with a language which approximates a natural language, systems simple to operate and designed for rapid assimilation by an untrained user. The use of different computers should be provided, beginning with minicomputers, without interfering in their operating system, and the use of various peripheral devices should be provided. The descriptions of user's problems compiled for computers of the same type should be suitable without any changes at all for other computers. The system should be open, should permit free build-up, flexible unified integration with the user's

problem-oriented software, and should provide connection of different applied programs and their modules. Of course, the system should have access to actuating devices such as machine tools with program control and so on.

We note that there are bases for this type of system that are rather simple, developed within 1-2 weeks and which meet the above criteria. These investigations are in the sphere of observation of one of the technical committees of the USSR National Committee on Automatic Control.

It would be interesting to consider the problem of what a common base, the "core" of automated design is to which the program modules, optimization programs and so on should be connected. This would facilitate development of a common, and to a significant degree unified SAPR.

We have already noted that the increasing laboriousness and scales of design become an obstacle for further expansion of ASU utilization, making it necessary to automate their design and programming. These investigations are being conducted both in the USSR and abroad. Part of the organizations is proceeding along the path of standardization--development of standard applied program packs, each of which corresponds to solution of one or another problem. This group also includes development of standard program modules.

Standardization of programs and program modules is undoubtedly a useful, but slowly proceeding process. The efficiency of standardization depends on the identical nature and stability of objects, but expenditures for tying in standard design solutions are frequently comparable to expenditures on the original design.

Talking about the stability of controlled objects, we note that the flaw of many concepts of design and utilization of ASU is concealed here. It turns out that most ASU are considerably more dynamic than is usually assumed. Control objects are modified during prolonged design and debugging and are improved during operation and their form is changed, including their structure and the format of the accounting documentation. Many examples can be cited when requirements on a system change' as a result of many years of design and it was obsolete at the moment it was introduced. The inflexibility of hardware and software places an obstacle in fr of making the necessary changes and postulation of new problems, and even more so if the user has no qualified programmers.

Should one be amazed by the fact that if the user has no flexibility of programs, he loses interest in the ASU or that "psychological resistance" even occurs? Not so much the users and the "user" qualities of ASU are guilty in this case. Based on this, one should develop a new approach to automation of design in order to fulfill the following requirements:

the speed of design of an ASU with the simplest method of preparation of the input information and reduction to a minimum of the length of the analysis-development-design-implementation cycle;

orientation to the user, provision of the capability of loading the system with problems in the order of their decreasing efficiency, of modernizing the system and inclusion of new problems in it through one's own efforts, without inviting "intermediaries" and highly qualified programmers;

direct automatic printout of the control programs by the system itself.

Solution of these problems is quite realistic and can be confirmed by the examples of systems which meet these requirements.

Throughout the world, qualified specialists study production processes from the viewpoints of control, investigate the properties of mathematical models of processes and develop ever more effective control algorithms. This is tedious and expensive work. To facilitate the utilization of these scientific results, scientists strive to classify control objects and to standardize on this basis models of them and methods of control.

It would seem that when designing a specific ASU TP [Automated production control system], it would be sufficient to take the corresponding scientific material from the enormous arsenal and to begin design. Unfortunately, this frequently cannot be done. The reason is that objects which are typical at first glance, are not identical in practice, require additional investigation and alter their characteristics during operation.

In order to cope with this and with similar factors, one of the universal principles --adaptation--is employed. As is known, adaptation theory has achieved extensive development. One of the implementations of this principle was a control system with identifier in a feedback circuit. This system, executed in the form of a computer with set of sensors and outputs to actuating mechanisms, is connected to the controlled object. It "observes" it, constructs a model of the object within itself, selects its parameters and is thus gradually taught. When the model with all parameters of the object is built-in, the system changes to the control stage, at the same time refining the parameters of its own model. Many years of operating experience at a pipe-rolling plant proved its effectiveness and reliability. The system has a number of valuable properties. It frees the creative and physical efforts of man and reduces the length of design and debugging.

The concept of the system has now been realized in different countries (the mentioned system in the USSR was apparently the first in time). This concept is now being utilized in the United States, Sweden, West Germany and other countries. We feel that this concept is one of the very promising trends, and even more so since it is applicable to control of facilities of the most diverse physical nature.

An interesting trend are teaching programs, by means of which problems of large dimensions can be solved on the basis of nonclassical methods, automatically utilizing the parameters of similar problems of lesser dimensions.

Let us now dwell on the jumps related to the appearance of new hardware. An important "intermittent" trend in the entire world is microprocessor systems and local networks of microcomputers. This trend is unusually broad and it relies to a significant degree on technology, while the control problems rely on development of the theoretical fundamentals of constructing these systems.

The use of microprocessors may in the near future lead to fundamental changes in control systems. During the past 30 years, control theory has proposed a set of complex and precise optimum control algorithms, while in practice simple, but

nonoptimum laws of regulation were mainly used due to the difficulty and uneconomical nature of realizing them.

The use of microprocessors, microcomputers and programmable controllers as direct digital regulators, the behavior of which is determined by the program stored in the memory of the microcomputer, permits an expensive realization of algorithms of any complexity in distributed production control systems, which stimulates further theoretical investigations.

Decentralized data processing systems with distributed configuration based on local microprocessor networks or local microcomputer networks have a number of advantages such as high viability, flexibility and the capability of reconfiguration. They permit qualitatively new solution not only of many problems of technical control, but also a number of problems in organizational systems.

Let us dwell somewhat on a trend related to the appearance of new hardware, namely, on problem-oriented control computer complexes with nontraditional structure. This is a family of computers of type PS with homogeneous multiprocessor structure automatically rearranged during problem-solving. The family of PS consists of several models of different calculating power and different specialization.

Let us assume that most readers are familiar with this problem, already realized on small models of PS serially produced for control of production processes. Higher-end models will be serially produced during the current and subsequent years. A geophysical computer complex described in this issue of the journal has been constructed on the basis of one of the higher-end models of PS. I feel that an important future belongs to machines with rearrangement structure in development of multiprocessor highly productive failure-free fourth-generation control computer systems and that they are undoubtedly related to the jump-ahead phenomena in our science and technology.

Which of the enumerated intermittent phenomena are actually "growth points" and which of them will lead to qualitatively new changes in the science of control and which will point the way to the future.

After all, forecasting is a difficult task and of course the indicated list does not encompass all the trends of our science and it primarily does not indicate the development of a wide range of fundamental research in the field of control theory. However, it is impossible to fully encompass these problems in one article.

In conclusion, a few words about a much broader control problem, unfortunately unformalized and consequently unsolved even by using the most powerful computer. This is the problem of management in the national economy and namely: how to place enterprises and other organizations into conditions in our rigidly planned economy when most of them will aspire to maximum rates of technical progress?

This problem, which we have faced for many years, and specifically mentioned in the author's article in PRAVDA, dated 20 March 1980, goes far beyond all that has been mentioned in its significance.

It is pointed out in the indicated article that technical progress, necessary on the whole to the national economy, is of insufficient interest to individual links of

the science-national economy chain. Of course, implementation of the decree of the CPSU Central Committee and the USSR Council of Ministers "On improvement of planning and intensification of the effect of the economic mechanism on increasing production efficiency and work quality" will improve this situation in time.

The difficulty of solving the problem includes the fact that we do not adequately take into account the role of man in the control system. Moreover, every person has his criteria of behavior and his own "scale of values." The main thing for some is acquisition of something in the material or prestige plan and the danger of losing this; in others it is a quiet life with minimization of labor; and in still others it is an aspiration toward creativity and enthusiasm. These people simply cannot live otherwise and they must create something new. Technical progress is based on these very enthusiasts of the most diverse hierarchical level and is to them that we are obligated for the greatest scientific and technical achievements, since there are no stimuli in our economic mechanism which automatically accelerate technical progress.

It was mentioned above that increased quality at the given level of technology also requires increased labor expenditures, exactly the same as an increase of product quality. Consequently, both one and the other is related to an increase of labor expenditures. And what does one do with the capacity of an enterprise fixed and with utilization of available reserves?

Thus, quality is in contradiction to quantity. What is the solution? Only in technical progress and in new scientific and technical ideas which advance a product and the enterprise to a higher level!

We feel that with the existing system of management, the basis of the indicated difficulties is a common factor: this is the actual many years of underestimation of technical progress, improper consideration of its position in the common "scale of values" and failure to consider the main laws of control. Quantity is frequently regarded as of paramount importance at different levels in planning, while problems of quality and of new technology are placed in second and third positions. This generates "lazy thinking" and leads to the temptation to copy foreign technology, which in turn predetermines our lag.

A few more words about man. With all the difference of natures, most people cannot be indifferent how society and managers evaluate their work; therefore, they act, taking into account to one or another degree the "scale of values" of managers, considering what they will be praised or criticized for and for what their bonuses will be increased or reduced.

Based on this, under our modern conditions, without touching on serious transformations, the most effective would be the following: quality and scientific and technical progress rather than quantity should be decisively moved to the forefront in planning, operational control and stimulation. In the final analysis, it is this that determines the improvement of a product and labor productivity. The measures which create an acute self-interest and acceleration of scientific and technical progress also primarily determine the prospects for accelerated development of our national economy.

Of course, evaluation of scientific and technical progress is a difficult matter and psychological as well as scientific and technical difficulties are encountered

here. Nevertheless, approaches and solutions can be found with serious postulation.

Problems of management in the national economy with regard to common laws of control theory and problems of man in the control system, of evaluation criteria and of the structures of the system, as before, remain the most important problems awaiting their solution during the 11th Five-Year Plan.

6521
CSO: 1863

UDC 65:681.3"71"

TOWARD NEW POSITIONS OF AUTOMATION OF CONTROL IN BASIC SECTORS OF INDUSTRY

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 9-11

[Article by Candidate of Technical Sciences M. P. Babin, chief of All-Union Production Association Soyuzpromavtomatika, and Candidate of Technical Sciences A. A. Levin, chief engineer of VPO Soyuzpromavtomatika]

[Excerpts] The expenditures for individual components for development of ASU TP [Automated production control system] are now evaluated in the sectors of industry under consideration by approximately the following relations:

applied software (mathematical modelling, control algorithmization, design and machine programming) comprises 30 percent;

acquisition of measuring and control apparatus comprises 30 percent;

acquisition of control computer complexes--UVK (including devices for communicating with the object and operating personnel) comprises 20 percent;

performance of installation and adjusting work comprises 20 percent.

6521
CSO: 1863

UDC 681.3.004

PROBLEMS ORIENTATION OF COMPUTER COMPLEXES OF THE INTERNATIONAL SMALL COMPUTER SYSTEM

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 20-23

[Article by Doctor of Technical Sciences G. I. Kavalerov, USSR deputy minister of instrument making, means of automation and control systems]

[Text] The most important features of small computers (mini- and microcomputers) which determine their widespread and mass use are accessibility to the user (in cost, overall dimensions and programming and operating simplicity) and the capability of adjusting information processing and automation of control of various types of facilities to specific problems.

Computers of this class, as is known, provide effectiveness of automation of both complex and large as well as relatively inexpensive machines, units and scientific equipment and make development and use of individual ("personal") computer complexes economically justified for scientific workers, engineers, economists and specialists in the field of control.

Design of diverse decentralized, distributed and network structures, feasible distribution and the use of the calculating capability of large universal computers and design of hierarchical control systems are provided when using small computers. This in turn determines the very extensive diversity of requirements which are placed by specific spheres of application on computer equipment of the given class.

Detailed analysis of the broad spectrum of diverse applications of minicomputers was made with regard to these features of small computers even during the first stage of development of the SM EVM [International small computer system] and the priority areas of application were selected in which the hardware and software of the SM EVM make it possible to achieve the greatest technical and economic effect.

These areas include control of production processes and production equipment in industry (ASU TP), automation of design in industry and construction (SAPR) and automation of scientific experiment (SANE).

All the principal engineering and organizational decisions on configuration of the computer, block-modular design, selection and standardization of interfaces and structures, nomenclature of hardware and composition of software and specialization, cooperation and development of SM EVM production were adopted with this orientation to these areas of application.

The SM EVM models of the first unit produced in large series have sufficiently high productivity (up to 800,000 operations per second), expanded OZU [Internal storage] capacities (up to 128K words) and a wide nomenclature of peripheral devices including modular devices for communicating with the facility [1-3].

The adaptability of the computer complexes of the SM EVM to user problems in different areas of application is provided by:

the unit (modular) principle of designing all the hardware with provision of systems compatibility;

efficient design of a number of peripheral devices and functional blocks with characteristics which provide continuity for the user during improvement of them and during development of the system;

developed general and specialized systems software for different configurations of computer complexes and their operating modes (10 varieties of operating systems, software for organization of remote processing, for multimachine complexes, debugging equipment, test programs, drivers for control of nonstandard peripherals and so on);

development of specialized intrasystems communications equipment and specialized peripheral devices (for example, USO [Unified organization systems] and CAMAC controllers, graph plotters, information readers, concentrators, terminal stations and so on);

continuity of software both with the preceding modular system of computer technology (ASVT-M) and with development of the SM EVM.

Rather extensive experience in development and use of control and information computer systems for different objects has now been accumulated.

The total number of computer complexes based on ASVT-M models (the M-6000, M-7000 and M-400) and the SM EVM exceeds 10,000 units. The volume of accumulated user software is estimated at not less than 700 million instructions. More than 12,000 operating personnel and programmers have been trained.

Achievement of the greatest efficiency for the mass user of small computers by means of problem orientation of them under conditions of the growing scales of production and application of SM EVM in the national economy is the most timely problem and main feature of technical policy in development and production of SM EVM.

The materials of questionnaire examination of ministries and agencies were analyzed to evaluate the prospects for 1981-1990 for using computer equipment of this class, to determine the most effective spheres of application and to systematize the user requirements. Economically feasible use of mini- and microcomputers for automation of control and information processing at different facilities of the national economy was evaluated from the results of the analysis, the readiness of facilities for effective automation was determined and the number of these facilities and the required number of computer equipment was estimated.

According to data of sectors of industry, approximately 20,000 highly productive production processes, powerful units and plants will be operating by the year 1990, for which design of ASU TP using computers is economically feasible. The potential need for automation using computer equipment is estimated at several tens of thousands of facilities in the nonindustrial sphere (transport, communications, construction, agriculture, health and so on). The need is estimated at approximately the same number of facilities in the sphere of automation of engineering design, automation of scientific experiment, testing of industrial products and programmed occupational training and retraining of workers.

Moreover, automation of various machines, devices and apparatus with local (decentralized) information processing requires a large number of built-in microcomputers and microprocessors. There are now some preliminary data of sectors of industry on automation based on built-in microcomputers of rather large-series equipment. According to literary sources, thousands of varieties of applications of microprocessors and microcomputers are operating in foreign practice.

The indicated materials permit a number of priority fields of SM EVM application to be allocated for the periods 1981-1985 and 1986-1990 with regard to the increasing computer equipment resources and accumulated experience in automation of facilities of different complexity and designation.

They include:

transport--control of switching stations on railroads and cargo handling at seaports, river ports and airports, control of passenger and baggage registration, of the ticket sales system and railroad traffic, control, monitoring and engineering diagnosis of transport facilities and so on;

commerce and material and technical supply--management of warehouse complexes and goods bases, information-monitoring systems of large commercial enterprises;

health--operational physician checking of patients in clinics, diagnostic systems, automated management of medical histories and control of diagnostic and therapeutic equipment;

agriculture--control of the production processes of large animal husbandry and poultry-breeding complexes, combifeed production, greenhouses, enterprises for afterharvest processing and storage of grain, cotton, sugar beets and other crops, management of water distribution networks and information systems of the agrochemical service;

training--occupational and retraining of personnel (simulators, automated classrooms and examination modules and reference systems);

environmental protection--automated systems for monitoring the state of the atmosphere and reservoirs.

Moreover, a number of effective trends in use of SM EVM will achieve significant development in previously adopted (traditional) fields of application (ASU TP, SAPR and SANE): in machine building--control of complexly mechanized sections (machine tools with ChPU [Numerical program control] and robots), machining and

automatic lines, galvanic production, product quality control benches, preparation of programs for machine tools and sections with ChPU and so on; in the food industry--control of production processes at creameries, slaughterhouses, locker plants, canning plants, mills and elevators and confectioners; in construction--extensive use of methods and hardware for automation of design in housing, civil, industrial and highway construction; in prospecting and field geophysics--automation of processing information flows at field stations and at regional centers; in development of local and distributed computer networks (of different classes), preparation of information and remote data processing with organization and development of collective use of computer equipment.

The experience of design and use of simple and complex control and information computer systems, on the one hand, and analysis of the planned trends in use of computer equipment, on the other hand, now make it possible to formulate a rather well-founded scientific and technical concept of small computer development in the country. The concept developed on this basis, which corresponds to the real, specific needs of the national economy and available material resources, is free of copying foreign prototypes and relies entirely on domestic scientific potential.

(This of course does not mean that foreign experience is not studied and not analyzed).

The most important feature of this concept, as already indicated is developing methods of providing the greatest adaptability of hardware to problems of specific users from the position of minimization of expenditures with maximum efficiency of the equipment used.

The main contradiction which must be overcome with extensive introduction of control computer complexes is retaining the advantages of mass production (production of a limited set of devices and modules) with simultaneous fulfillment of the requirements of different users, i.e., provision of economy of funds and time in development of specific control systems.

The indicated contradiction within the SM EVM is solved both by increasing the technical level of the models of all devices themselves contained in the SM EVM and in improving their basic nomenclature and technology of combination of hardware and software in developing specific configurations of control computer complexes (UVK).

It is appropriate to recall that UVK for any purposes were initially produced with rigid or low-developed structure (Dnepr and UM-NKh). Output of more than 30 varieties of standard complexes (M-6000, M-7000 and M-400) of different structure and with different composition of equipment, differenting more than tenfold, was then organized on the basis of modular standardized systems and modules of ASVT-M. The capability of building up hardware during development or step by step introduction of the control system with the user was provided.

Complexing and output of specified UVK by order for the project of a specific control system were organized during the third stage for complex and crucial objects. The entire necessary and sufficient set of hardware (of the composition of the ASVT-M or SM EVM), systems (general and special) software, adjusted and debugged for a given configuration, and a set of test programs and control problems are delivered by the manufacturer to the customer. The applied software is developed and debugged by the customer or designer. However, difficulties and contradictions

between the number of orders of specified UVK and the capabilities of plants occurred in this method. Moreover, subsequent development of the applied software for each system is managed individually to a significant degree and requires large expenditures of highly qualified labor.

It has been established that the equipment volume of purely computer devices (processor-memory and microcomputers) comprises from 12 to 1 percent of the total equipment (and is even less in cost) in relatively developed systems of this type. The dominating part in cost and equipment is occupied by USO, external memory, information input-output devices, intrasystems communication equipment and software in complex UVK. Systems, equipment engineering and operating problems occur here.

The following solutions have been adopted in development of specified UVK (for facilities where this is feasible) in the SM EVM:

a) minimization of the number of configurations of computer devices contained in the SM EVM since this determines the number of varieties of hardware and methods of combining them;

b) standardization of the peripheral nomenclature which provides combination of specific configurations of computer systems based on processes of different configurations; this problem has two components: standardization and ordering of the combination of specifications of peripheral devices and standardization of their interfaces;

c) development of means and methods of programming which provide "adjustment" of a structurally and logically configured complex to the required range of functions (operating systems, program drivers, checking and diagnostic programs and means of automatic generation of program systems);

d) provision of the prolonged existence of the hardware system in production and use; the length of existence and development of the SM EVM system does not mean "freezing" the technical characteristics of individual devices; improvement of them should be continued intensively, but with adherence to the conditions of systems compatibility and continuity.

The first level of complexing and the basis for development of standard, specialized and problem-oriented (see below) UVK are series of all the required peripheral devices having outlets to standardized interfaces within the SM EVM.

Delivery of specified UVK configured from modular devices and modules in arbitrary combinations is related to the need to finish and check many versions of hardware and software configurations and to organization of complex interaction of plants and users.

Therefore, it is necessary on the one hand to automate the process of developing specified UVK and on the other hand to simplify the problem, breaking the UVK down into a number of functionally complete subcomplexes with simple hardware and software contacts between them. This retains essentially all the advantages of the unit-modular principle of system design and simplifies production and operation of UVK.

The second level of complexing is design and use of functional subcomplexes in the UVK.

The organization of subcomplexes of different designation (the energy block control console, the cashier's position in a ticket reservation system, the designer's position and so on) from peripheral equipment of various groups (displays, ATsPU [Analog-digital converters], NML [Magnetic tape carrier], NMD [Magnetic disk carrier], USO modules and so on) is accomplished on the basis of microcomputers.

Thus, structural decentralization of information processing and control of peripheral devices within the UVK is introduced. Part of the functions carried out in existing small computers by central processors is transferred to subcomplexes on the basis of microprogrammed controllers and microcomputers. Analysis shows that this method is especially effective and necessary in computer equipment of this class.

Functional subcomplexes can be divided into:

basic computer complexes (BVK) which include processors, the main internal storage, external memory and main systems input-output devices (the equipment within complex BVK may also be grouped into a number of systems subcomplexes with microprogrammed controllers);

terminal subcomplexes (TsK) connected to BVK by intrasystem or remote communications lines and containing all the required peripheral devices (except those which are contained in the BVK).

The capability of connecting terminal subcomplexes to two BVK is provided to ensure high viability and flexibility in design of multimachine complexes.

The terminal subcomplex is either a multifunctional peripheral device or a group of peripheral devices with respect to the BVK.

The terminal subcomplex can be delivered to the user as a finished article having hardware and microprograms, including test-diagnostic programs. The capability of connecting terminal subcomplexes of SM EVM to BVK of other systems (ASVT-M, YeS EVM and Elektronika) through any of the interblock interfaces used in the SM EVM is provided. TSK can be divided by functional designation into the following groups: terminals for communicating with operating personnel, USO terminals, subcomplexes for entry and recording of text and graphical information, communications subcomplexes and subcomplexes for communicating with other systems.

Some TSK can be directly related to two or more groups. All the peripheral devices (not included in the BVK) are joined into TSK based on a microprogrammed controller through a standardized interface IUS. The devices are controlled by the microprogram.

This subcomplex is connected to the machine systems interface of a computer of any configuration by a coordinator whose load is also realized by the microprogram. In this case the composition of the subcomplexes is invariant to the processor configuration. The determining factor here is selection of the microcomputer (the basic instruction or microinstruction set, type of peripheral interface and so on).

Production of microprogrammed controllers of the SM EVM and SM-1800 microcomputers designed to implement the indicated functions is now being organized.

The configuration of peripheral equipment in the form of TSK yields the following main advantages compared to traditional structures:

a) configuration of specialized UVK is considerably simplified, generation of the operating system and the control problem at the center of the complex depends to a small degree on the composition of peripheral devices; the collisions which usually occur when working with nonstart-stop devices become impossible due to the buffer storage of data in lower subcomplexes; a check of numerous structural versions on a test area is not required and the configuration of territorially dispersed complexes is especially simplified;

b) operation of the UVK is facilitated, primarily due to the capability of autonomous checking and diagnosis of the subcomplexes;

c) it is no longer necessary to develop drivers for all combinations of operating systems, peripheral devices and methods of connecting them;

d) the time and memory of the central processor is conserved significantly and its effective productivity is increased;

e) the equipment controllers of peripheral devices are simplified;

f) it becomes possible to organize self-contained production and independent delivery of subcomplexes to the user while reducing the cost of the entire computer system.

Development and use of specialized processes oriented toward specific classes of problems in the BVK occupies a special position in solution of problem orientation of the UVK, i.e., their maximum adaptability to problems of various users. Without dwelling on this well-known problem, let us note that a number of special processors: digital arithmetic, fast Fourier transform, matrix processors, mega-mini-computer processor and so on, has already been developed in the second unit of SM EVM equipment.

Based on basic complexes of SM EVM, TSK, problem-oriented processors and multiprocessors, specialized UVK will be configured for specific applications during the next five-year plan. Systems engineering, engineering and organizational methods of complexing in large-scale production is now being worked out with organization of deliveries of specified SM-2 and SM-4 complexes.

Fulfillment of this task requires development of a modular software system and accumulation of applied and service program packs which provide complexing in all directions.

Thus, considering the prospects for development of small computers, one can state that the main problems are not so much in development of new computers as in development of a wide nomenclature of the equipment which makes up terminal networks and development of methods of complexing (systems engineering, engineering and organizational) which provide large-series production of UVK. Development of means and

methods of programming which ensure the procedure of configuration and setting up of user programs in complex problem-oriented UVK is required.

Consideration of future trends and fields of application of small computers shows that, besides the relatively complex computer systems for many priority facilities equipped with specialized UVK, a large number of computer complexes based on mini- and microcomputers oriented toward specific classes of facilities combined by community of information gathering and processing technology rather than toward individual specific control or information processing systems is required.

The concept of developing problem-oriented software-hardware complexes (POK) of the SM EVM which makes it possible to overcome the contradiction between retaining the advantages of large-series production of standard complexes and subcomplexes and expanding the number of types of facilities and their application with minimization of time and monetary expenditures for development of specific systems was formulated by the Council of Specialists under the supervision of Corresponding Member of the Ukrainian SSR Academy of Sciences A. A. Stogniya within the framework of developing the second unit of the SM EVM.

The efforts of ASU developers, hardware developers and manufacturing enterprises are combined during the early stages in development of POK to achieve a common goal--development of highly effective control systems in selected fields of application.

Unlike standard and specialized UVK, the main applied software which permits the user to solve the main problems with minimum finishing of programs is developed and delivered in the POK.

When developing a POK, it is important to determine the rather representative class of facilities and problems which is encompassed by given POK. The level of POK circulation in specific user complexes (PK) determines the extent of problem orientation and consequently the efficiency of POK and PK.

A POK is a set of hardware and software and methodical, design and normative and technical materials for realization of a given set of tasks for automation of a specific class of facilities joined by common information processing technology, unanimity of operating modes and operating conditions. The information processing technology is determined by the sequence and specifics of data processing in the stages of preparation, gathering, preliminary and content processing, formulation, printout and utilization of the results. Thus, a POK is developed as a base (in other words, a set of "semifinished product manufacturers") for development of systems for specific objects--PK.

It must be noted that the idea of developing software-hardware complexes oriented toward solution of specific classes of problems was advanced by Academician V. M. Glushkov at the end of the 1960s, but it did not achieve the proper development at that time.

Development of PK based on POK can be accomplished by synthesis of:

POK modules and subcomplexes with possible parametric adjustment for specific conditions (not all modules of a given POK can be connected to PK);

selected POK with additional hardware and software modules specific for a given PK;

several POK into a unified PK on the basis of a single more powerful processor;

several POK into a unified multimachine PK for complex ASU and data processing systems.

The idea of POK development found practical realization in development of complex automated operators' positions (ARM) and measuring computer complexes (IVK) for automation of design and for automation of scientific research. The development of POK of the SM EVM for ASNI [Automated scientific research system] was preceded by systems analysis of the requirements of a number of institutes of the USSR Academy of Sciences [4]. Specialists of the Council on Automation of Scientific Research attached to the Presidium of the USSR Academy of Sciences and specialists of INEUM [Institute of Electronic Control Machines (of the USSR Academy of Sciences)] --the head organization on development of the SM EVM--participated in this work. Based on examination of the leading institutes of the USSR Academy of Sciences, the most important areas of application of SM EVM were determined, the classes of automation systems were refined, the requirements on hardware and software were determined and the general principles of PK development for specific applications were refined. A series of the eight most important IVK (IVK-1--IVK-8), which are now produced by plants, were formed as a result.

Important practical problems were solved for an entire class of IVK--this is "tying-in" nonstandard USO, development and introduction of controllers of CAMAC and ASET complexes and of nonstandard peripherals (the N710, N711 and other graph plotters) into serial production.

Moreover, problems of refining the required composition of modules of the CAMAC complex, development and serial assimilation of a number of modules and solution of problems of providing IVK with a complete composition of modules were solved.

Finishing the software in development and inclusion of the necessary drivers of devices, the monitor of the CAMAC complex, set of test and metrological programs and so on into operating systems was also common problems for development of IVK. The indicated finishing of hardware and software solved the most general problems and provided relatively wide orientation toward application of SM EVM for automation of scientific research. Problems orientation was intensified by developing applied program packs for processing experimental data, for example, common packs for some biological investigations, geophysics, crystallography, electronics, solid-state physics and so on.

The indicated examples, on the one hand, illustrate the complexity of problems solved in development of the SM EVM with orientation toward application in SANE, and on the other hand they indicate that the concept "problem-oriented complex" itself for a specific class of applications is a "sliding" concept and may be investigated as a function of the list of problems of specific PK encompassed by a given POK. In this case some redundancy of the implemented solutions at upper levels of problem orientation determines the broader capabilities of specific implementations of PK for given conditions of applications.

Emphasizing the effectiveness of a practically realizable problem orientation program, one can note that the institutes of the USSR Academy of Sciences receive expanded complexes with inclusion of the required nonstandard equipment supplied with finished operating systems (for example, OS RV [Real-time operating system]) and program packs which are not supplied in standard form in typical staff complexes by manufacturing plants.

According to the plan of SM EVM development, work is carried out to formulate orders to develop the most important POK on the basis of analyzing the areas of application and consideration of the materials of leading systems organizations to justify the feasibility of developing specific POK of SM EVM. More than 100 requests for development of POK were considered.

POK which encompass such fields of automation as follows were developed within the given plan:

a) automation of the shops of enterprises with digital production (in automation of group machine tool control, production of programs which control machines with ChPU, monitoring of product output and accounting for the idle times of equipment, control of suspended conveyors systems with accumulation, sorting and automatic addressing of the load and dispatching of part of the machining with automatic control of the transport-warehouse system);

b) automation of analog and analog-digital production processes (specifically, automation of the production process in the section of rolled steel furnaces and control of the power regimes of the smelting process and calculationof the charge);

c) automation of the distribution of energy, transport and continuous material flows (specifically, automation of control of large blocks of electric power plants, dispatcher control in power engineering and control of the water delivery and distribution process in an irrigation system);

d) automation of scientific experiments--gathering and processing of experimental data and control (specifically, automation of gathering and processing the data of laboratory experiments based on active dialogue with the investigator, automation of monitoring and measurement on the basis of equipment of the CAMAC complex and with preliminary processing of information for general physics in the investigations;

e) automation of the medical service of the populace (specifically, automation of medical documentation management, analysis of the condition of the human organs during examination and treatment and processing of data of instrument examination of patients in therapeutic institutions).

The work on analysis of promising applications of SM EVM during the 11th Five-Year Plan indicates the possibility of meeting the needs of a number of leading ministries by using the POK of SM EVM and it is proposed that the greatest number of PK based on the POK of SM EVM be developed in the sphere of equipment control, organizational control and control of production processes.

Thus, a considerable part of the objects of automation (from those indicated above) can be equipped with the hardware and software of the POK of SM EVM.

Conclusions

The effectiveness and mass nature of using small and mini- and microcomputers in the national economy to automate information management and processing are determined to a significant degree by the relative inexpensiveness and flexibility and the adaptability of computer equipment of this class to different user problems. The small computer system was developed and is being further developed on the basis of study and systemization of the real needs of the national economy with orientation toward specific priority areas of effective utilization.

2. The problem orientation of computer complexes of the SM EVM is achieved at different levels of development of the system by:

developing a number of standardized modular functional devices and modules with provisions of systems compatibility and continuity;

working out developed general and special systems software oriented toward different modes and procedures of information processing and interaction with peripheral equipment (standard and specialized);

conversion to a decentralized structure of information processing and control of peripheral devices with division of UVK into functional subcomplexes and basic and terminal complexes managed by programmable controllers and microcomputers;

delivery to the user of a number of standard computer complexes (with wide range in the set of hardware and software) and of specified UVK with composition according to the design of the system;

development and delivery to the user of a number of self-contained TSK of different designation;

development of a number of specialized processors and multiprocessors oriented toward specific information processing algorithms;

development of problem-oriented programming languages, versions of standard languages for specific classes of applications, program debugging equipment, problem-oriented general-purpose applied program packs and also creation and development of a stock of applied program packs for the SM EVM;

development and delivery to the user of software-hardware complexes, computer complexes and problem-oriented complexes for solution of specific standard information management problems and information processing.

BIBLIOGRAPHY

1. Naumov, B. N., "The International Small Computer System," PRIBORY I SISTEMY UPRAVLENIYA, No 10, 1977.

2. Rezanov, V. V. and V. M. Kostelyanskiy, "The SM-1 and SM-2 Control Computer Complexes," PRIBORY I SISTEMY UPRAVLENIYA, No 10, 1977.

3. "The International Small Computer System," Catalog "International Exhibit of YeS EVM and SM EVM," 1979.

4. Kavalerov, G. I., "Measuring and Computer Complexes," PRIBORY I SISTEMY UPRAVLENIYA, No 11, 1977.

6521
CSO: 1863

UDC 681.3"313"

THE INTERNATIONAL SMALL COMPUTER SYSTEM--STATUS AND PROSPECTS FOR DEVELOPMENT

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 24-28

[Article by Candidate of Technical Sciences S. S. Zabara, deputy director of Scientific Research Institute of Scientific Work, Corresponding Member of USSR Academy of Sciences B. N. Naumov, director of Institute of Electronic Control Machines (of the USSR Academy of Sciences), Candidate of Technical Sciences V. V. Rezanov, deputy director of Scientific Research Institute of General-Purpose Computers on Scientific Work, and Ye. B. Smirnov, chief of All-Union Production Association Soyuzelektronmash]

[Text] The tasks posed by the 25th CPSU Congress on improving management of the national economy were a prerequisite of qualitative changes in the work of the subsector. Technical re-equipping of enterprises of the subsector was carried out at accelerated rates: an increase of production capacities provides output of essentially the same number of small computers during 1980 as was produced during all the years of the Ninth Five-Year Plan. The qualitative indicators of computer equipment (SCT) produced at Minpribor [Ministry of Instrument Making, Means of Automation and Control Systems] were improved. Together with essentially complete restoration of the nomenclature of small computers and peripheral devices produced, the volume of output of articles which meet the requirements of higher category of quality increased twofold during the five-year plan. Modern models of the SM EVM [International small computer system] are approximately an order greater than models of the ASVT-M [Modular system of computer technology] series by generalized technical and economic indicator.

All conditions were mainly created during past years so that small and microcomputers became the most mass means of automation of complex processes in all sectors of industry, in scientific research and also in numerous applications which touch the nonindustrial sphere of the national economy (health, education, commerce and so on. New effective methods of information processing (machine graphics, real-time digital signal processing and development of remote data processing equipment at computer complexes and networks) find a place as ever newer areas of SVT application are developed. Improvement of the cost-productivity indicator achieved in SM EVM models permits a high level of SVT integration with automation facilities to be achieved. The developers of automatic manipulators, equipment with ChPU [Numerical program control] and industrial automation equipment are faced with especially important tasks in this regard.

The leading fields of SM EVM application are systems for control of digital and analog production processes, systems for automation of scientific experiments (SANE) and also complexes for solving problems of design automation and problems of the organizational and management link. The range of applications determines the need to solve a complex of important scientific and technical, production and economic problems on a government and international scale.

The most important among these problems should be regarded as:

achievement of standardization of hardware and methods of complexing (standard interfaces);

assimilation of new methods of designing basic and problem-oriented complexes which permits implementation of the design configuration of specialized complexes on large scales and circulation of them;

achievement of program compatibility of SM EVM models and their continuity with respect to ASVT-M models, accompanied by a complex of investigations to standardize means of programming on CEMA scales;

standardization of design and production decisions at the level of international standards which provides a high level of SVT quality, the possibility of developing intersector and international cooperation and conditions for conversion from large-series to mass production of SVT;

working out unified requirements on the promising microelectronic component base;

organization on country-wide scales of a network of installation and adjustment organizations and warranty servicing centers.

Main Principles of SM EVM Development

The need to develop SM EVM within compressed deadlines determined the use of new methods of organizing the work [1]. Work usually performed sequentially was combined for the first time in the USSR in development of small computers: development of hardware and software and design of problem-oriented complexes (POK), which was carried out simultaneously with development of promising microelectronic components by the electronics industry, mass application of which in SM EVM made it possible to significantly increase the technical level of the articles. The structural optimization of families of SM EVM models was technically realized in the following manner: the variant modules are the central processors and in some cases the controllers of devices, whereas the main set of peripheral devices and intermachine communication devices is common for all models.

The basic engineering solutions adopted in SM EVM satisfy the conditions of mass production, providing the capability of using automated and mechanized methods of production on the basis of leading technology. Extensive work on technological preparation of production was carried out to organize serial production of SM EVM.

Thus, for example, the process flow sheet for production of the SM-4 at the leading enterprise of the subsector--the Kiev PO [Production Association] Elektronmash

imeni V. I. Lenin--includes five ASU with the most crucial production processes and up to 40 automated systems for adjustment of the electronic blocks and quality control of the product at all the main stages of producing the SM-4 UVK. A total of more than 2,000 production processes was worked out during these years at enterprises of the association and 825 types of accessories and 25 new production benches were designed. The SM-3 and SM-4 computers were included in the control and checking production lines at enterprises of the subsector even during the second year of their output. SAPR [Automated design system] of printed circuit cards, control of the drilling line and automated checking are distinguished by high efficiency.

The "second circle" of producing computer and control complexes based on the SM EVM is acquiring ever greater significance. Thus, output of a number of problem-oriented UVK for different fields of application has been organized on the basis of SM-3 and SM-4 hardware and software. The Kiev Plant Tochelektropribor imeni komsomol Ukrainy, the Cheboksary PO Elektropribor and the Vilnius Plant of Electronic Measuring Equipment are cooperatively producing a number of IVK [Information-computer complex] used in scientific research organizations. The Moscow Experimental Plant Energopribor, along with basic SM-3 complexes, is producing specialized complexes based on the SM-3 and SM-4 UVK upon specified orders of individual customers. A number of plants of other sectors have organized production of POK based on the SM EVM (ARM-R and ARM-M).

Expansion of the scales of UVK production advanced to the forefront problems of accelerated development of technology for software production and improvement of the technology of mass introduction of SM EVM into the national economy. Solution of the indicated problems made it possible for the first time in our country to deliver SM EVM complexes with a rather complete set of operating systems and applied program packs to the customer from the very beginning of serial production of the new family of computers. Development and realization of the principles of problem orientation of the SM EVM make it possible to integrate in time in the most laborious processes of developing the hardware and control systems and to deliver hardware-software complexes oriented toward specific classes of application to the customer. This permitted a reduction of the period of introducing the SM EVM by one to two years in some case.

The characteristics of small computers are primarily determined by the range of areas of their application. Small computers, on the one hand, are used extensively for automation of relatively inexpensive units and scientific installations and make development of individual complexes (for scientific workers, economists, engineers and workers of the control sphere) economically justified. On the other hand, the equipment of SM EVM provides the possibility of designing decentralized, distributed network computer structures with coordination of the operation of a large number of computers. It became possible with the use of small computers to develop complex control systems for electric power plants, shops, monitoring-testing and experimental installations. Combining small and microcomputers in UVK with efficient distribution of functions between them and also the use of specialized processors makes it possible to bring the productivity of these complexes in solution of a number of specific problems up to the level of productivity of large and superlarge computers.

Thus, even this far from complete characteristic of the feature of small computers indicates extremely broad fields of their application and predetermines the

diversity of requirements which are placed by specific applications on SM EVM. Detailed analysis of a wide range of applications was carried out during the first stage of SM EVM development. This analysis contributed to concentration of efforts in the most important directions of SM EVM development.

Characteristics of the First Unit of SM EVM

Models of the first unit of the SM EVM differ significantly in their technical level from machines of previous generations while retaining program compatibility with subsequent generations (SM-1 and SM-2 with the M-6000 and M-7000 and the SM-3 and SM-4 with the M-400). The physical volumes of the complexes were reduced significantly (approximately by a factor of five), their cost was reduced by a factor of two and indicators of reliability were improved twofold. Speed when performing short arithmetic and logic operations reached a level close to one million operations per second. Efficient combination of operations in the processor, apparatus execution of most systems functions for interrupt processing and automation of adjustment of the complex to different modes provide, for example, up to 300,000 averaged real-time operations in the model of the SM EVM with the highest speed--the SM-4 (this characteristic was determined on the extensive material of realizing real-time systems). Development of complex control systems, introduction of optimization information processing algorithms and development and introduction of data bases require significant expansion of the capacity of internal storage and realization of multilevel fields of external storage devices. Effective operating modes with OZU [Internal storage] capacity up to 128K words are provided in models of the SM-4 and SM-2, which corresponds to the modern level of requirements of ASU TP [Automated production process control system], systems for automation of scientific research (SANI), SANE and most dispatcher control systems [2, 3, 6].

Peripheral Devices of the SM EVM

One of the central positions in the overall program of SM EVM development is occupied by investigations to develop and organize in serial production the peripheral equipment for mini- and microcomputers. The peripheral equipment comprises from 70 to 80 percent of the cost of the UVK and significantly affects the main specifications and operating characteristics of the ASU--productivity and reliability. On the whole the peripheral equipment of the SM EVM is characterized by wide nomenclature, determined by the wide range of application of the SM EVM, by the high requirements on compactness (built-in modules in many cases), by the increased requirements on reliability and ergonomic parameters due to mass introduction and by relatively low cost which should stimulate introduction of computer equipment into all sectors of the national economy.

The nomenclature list of peripheral equipment of the SM EVM now includes practically all the necessary devices for development of control systems in the main fields of application. The nomenclature of the SM EVM includes a total of more than 100 peripheral devices, not counting remote data processing equipment, devices for communicating with the facility (USO) and production recorders.

A functional series of magnetic storage devices, including cassette magnetic disks with capacity up to 30 Mbytes, a number of standard cassette and magnetic tape and floppy magnetic disk storage devices has been developed. The series of printers encompasses sequential (character-synthesizing) devices with printing speed of

100-180 characters per second and parallel devices with productivity of 500-900 lines per minute. The display equipment--alphanumeric and graphical displays, including intellectual, color and half-tone displays, is rather widespread. The group of operator-system communication equipment is represented by radio terminal stations, which includes alphanumeric black-white displays of two types (for 512 and 2,000 symbols), supplied with the necessary editing equipment and also a graphical display with functions which provide design of memory circuits. The possibility of configuring the work site of the technician-operator according to the functional requirements on the engineer's work site and with regard to the configuration of the system has been provided.

The group of equipment for communicating with the facility is a set of unitized modules (analog-digital and digital-analog converters, commutators, normalizers, signal simulators, portable commutators and group converters and code control and code information gathering modules) which provide information gathering and printout of information on objects equipped with sensors, local automation systems and actuating mechanisms of GSP [State System of Industrial Instruments and Automation Equipment].

The specific results, as yet insufficient, have been achieved in the field of developing graphical information input-output devices. Devices for semiautomatic entry of drawings (Figure 1) alone have now been developed and a number of drum and plotting board type graphical information printing devices, including those on various special types of paper, are now under development.

Extensive work has now been carried out within the SM EVM to standardize the specifications on different classes of peripheral equipment which determine their competitiveness on a worldwide level. Peripheral interfaces have been standardized, providing technical and systems interchangeability of peripheral devices and consequently the propsects for scientific and technical and production cooperation.

Organizations and enterprises have specialized within Minpribor on development and production of peripheral equipment. All the enumerated classes of peripheral equipment have now been developed and put into production. The complex program for development of these investigations for 1981-1985 envisions a further increase of the technical level and volumes of production of peripheral devices.

Investigations to improve peripheral devices on traditional principles of information processing are directed toward improving their technical and economic indicators, namely:

an increase of reliability, speed of exchange and density of recording for magnetic storage devices due to use of integrated magnetic heads and improvement of the magnetic coating of carriers;

an increase of resolution, brightness and color gradations and also of the "intellect" of information display devices;

an increase of reliability and productivity, reduction of cost, reduction of overall dimensions and reduction of noise level for impact printing devices;

expansion of the nomenclature, an increase of the degree of integration for USO and so on.

A great deal of attention has been devoted in the program to development of peripheral equipment on new principles of physics: optoelectronic storage devices, fluid and laser printers, liquid crystal and plasma panel display devices and so on. Considerable simplification of the equipment which provides communication of peripheral devices with the computer by extensive use of microprocessors and special "peripheral" BIS [Large intergrated circuit] is being planned.

The fraction of graphical peripheral equipment of all types, remote processing and intrasystem communication devices, USO and also magnetic carrier input-output devices is being increased in the volume of the entire nomenclature.

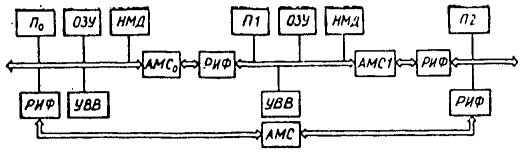


Figure 2

The principle of control decentralization is taken as the basis when developing SM EVM peripheral equipment. All peripheral devices are connected to the computer complex as part of systems or terminal subcomplexes configured on the basis of microprogrammed controllers and microprocessers. The microprogrammed controller designed for this purpose has already been developed and is being assimilated in serial production. This approach ensures the following main advantages compared to the traditional approach:

configuration of the specified complexes (especially territorially dispersed) and generation of operating systems for them are considerably simplified;

it becomes possible to organize autonomous production with complete checking for functioning and independent delivery of subcomplexes to the customer;

operation of the subcomplexes is simplified (primarily due to the capability to autonomously monitor and diagnose the subcomplexes);

the time and memory of the central process is conserved, the response of the system is increased with simultaneous decrease of requirements on the response of the central processor and unforeseen collisions which occur in existing systems due to simultaneous asynchronous operation of various devices are eliminated;

the apparatus required to connect peripheral devices is simplified and modernization of it is facilitated;

a significant reduction of the cost and overall dimensions and an increase in the reliability of the complex are achieved in most cases.

It has been suggested that systems subcomplexes be delivered as part of basic computer complexes, while terminal subcomplexes (intellectual video terminals, USO

terminals, data transmission multiplexers, data recording subcomplexes and so on) be produced in the form of self-contained articles with microprograms embedded in them which provide functioning in the given modes and checking and diagnosis. Development of the first of these subcomplexes is being completed.

The problem of developing UVK according to the individual plans of customers requires special attention. The practice of systems development showed that UVK cannot be produced and delivered in standard configurations like a general-purpose computer. Specified UVK (SUVK) were first supplied in domestic computer technology at the NPO [Scientific Production Association] Impul's imeni 25th CPSU Congress (Severodonetsk) in 1973 (the first three SUVK were delivered to the Kuznetsk Metallurgical Combine imeni V. I. Lenin). A large part of UVK is now planned in the form of SUVK from developments of NPO Impul's and VPO [All-Union Production Association] Soyuzelektronmash.

Examples of Means of Complexing in SM EVM

Special equipment and primarily an interprocessor communications adaptor (AMS) and bus switch--PSh [1, 3, 6, 8]--are provided in the structure and configuration of SM-3P and SM-4P processors to support design of multimachine and multiprocessor complexes. The interprocessor communications adaptor is connected to the "Common bus" mainlines of two complexes. The processor of each of the complexes may have access through the adaptor to any devices connected to the "Common bus" mainlines of another complex, having access to the program-control regions of the OZU. An example of complexing a three-machine highly productive conveyor type VK [Computer complex] using an AMS is shown in Figure 2 (here RIF is an interface expander--a device for building up the "Common bus").

The indicated AMS is the highest speed device for integration of computers that supports information transmission with delay of no more than 400 ns. Because of this, two-machine complexes based on AMS are used to parallel computer processes when the calculating capacity of one SM-4 computer is inadequate or when program overloading in the OZU is impermissible, design of load redundancy systems and high-response function sharing system during information processing.

Unlike AMS that provides direct communication between the OZU of two processors, another systems device--the PSh--makes it possible to connect a section of an additional "Common bus" from one processor to another by program signals (or with manual control--local or remote). An example of complexing a hierarchical UVK for concentrated facilities is shown in Figure 3.

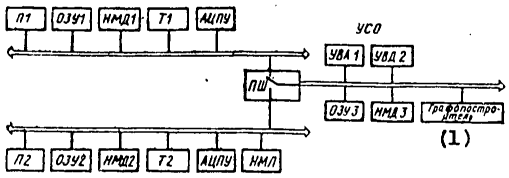


Figure 3

Key:
1. Graph plotter

A similar two-machine complex can be used in the following modes as a function of specific conditions of application.

1. Redundancy of processor P1. In this case processor P2 processes information in the background mode, periodically transmitting requests for use of the PSh in processor P1. If processor P1 fails, the PSh is switched to processor P2, which can continue real-time control of the facility. In this case the response of switching the computer operating modes can be increased if the OZU modules 3 are connected to the auxiliary bus.

2. Sharing of functions. In this case, for example, processor P1 provides operational information gathering, preliminary express analysis of information and rerecording of information to external storage devices as the OZU is filled up. Processor P2, achieving access to the PSh, processes information by the given algorithms within specific time segments.

3. An increase of the efficiency of utilizing expensive and unique peripheral equipment.

The structural capabilities of two-machine complexes are expanded if several AMS and PSh are used to create a multimachine complex with given set of functions.

SM EVM Equipment to Construct Remote Data Processing Systems

Development of the following equipment is provided in the SM EVM system to construct remote data processing systems (Figure 4): modems (M) and signal conversion devices (UPS), data transmission adapters and data transmission multiplexers (APD and MPD), automatic callup devices AVU and automatic callup adapters ADA, error protection devices (UZO) and also a number of terminals (T) of different composition and designation.

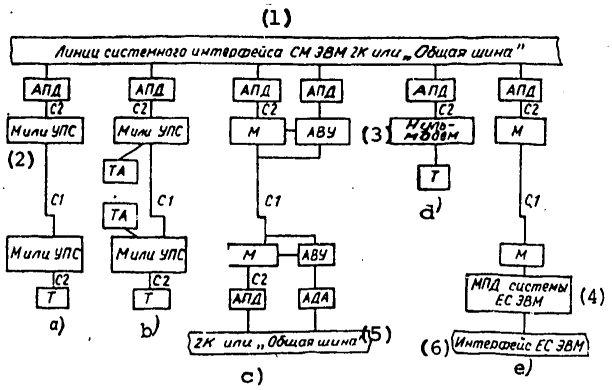


Figure 4

Key:
1. Systems interface lines of SM EVM 2K or "Common bus"
2. Mealy UPS

[Continued on following page]

[Key continued from previous page]

3. Zero modem
4. Data transmission multiplexer of YeS EVM system
5. 2K or "Common bus"
6. YeS EVM interface

The modems and UPS are designed to convert binary information coming from the UVK to a form which is required for transmission over communications lines and also for reverse conversion. In those cases when the terminal is in the immediate vicinity of the UVK, a variety of modems--the zero-modem--is used. The adapters and MPD are designed for connecting the modems and UPS to the systems interfaces. Unlike the APD, the MPD usually provides an outlet of the UVK to several communications channels.

Several versions of the possible structures of data transmission systems using SM EVM equipment are shown in Figure 4:

communication of the UVK with a remote terminal (Figure 4, a);

communication of the UVK with a remote terminal with the capability of conducting service telephone conversations through the connected telephone equipment TA (Figure 4, b);

communication of two UVK over allocated or commutated telephone channels with a capability of automatic establishment of the connection (Figure 4, c);

communication of the UVK with a nearby terminal (Figure 4, d);

communication of the UVK based on SM EVM equipment with the complex of the YeS EVM system (Figure 4, e).

The Software of the SM EVM

The instrument base of the modern technology of development and production of software and output of program documentation has been created as a result of realizing the first unit of the SM EVM. According to the general concept of problem orientation of the SM EVM, the production instrument equipment for software design also has problems orientation and is constructed in the following manner [1-3, 7].

The bases of program modules which formulated with regard to specialization and namely the following modules comprise the basis of instrument devices:

for logic-mathematical processing of data which includes programs of mathematical statistics, numerical analysis, optimzation methods, network planning and control and so on;

data technologies which encompass programs that provide data gathering and preliminary processing, input of data of different format, data transmission over communications lines, organization and servicing of data structures, information display and so on;

servicing and control of peripheral devices, syntactical and semantic expanders, intersystems communication devices and so on;

monitoring and diagnosis of hardware;

organization and control of the calculating process;

those which simplify communications of the user with the system.

The next level of instrument devices is programs for operating system (OS) generation for given applications and configurations of hardware. Generation of OS of the following types is provided in the first unit:

general-purpose for preparation and debugging of programs (punch tape and disk);

real-time (multiprogram paper tape and disk systems for small configurations, systems for control of rapid processes, multiproblem systems of broad designation for control of complex processes and facilities in real time);

with time-sharing--for solving information-logic problems and problems of a calculating nature in the collective-use mode;

for complex adjustment and testing of hardware.

Programming systems which provide preparation and debugging of programs in the interactive mode with ASSEMBLER, MACROASSEMBLER, FORTRAN, BASIC, BASIC-PLUS, DIASP, DIAMS and COBOL languages function under the control of the OS. A set of procedure and problem-oriented PPP [Applied program pack], which include numerical analysis programs, mathematical statistics programs, network planning and control methods, methods of servicing USO and remote processing equipment, machine graphics, management of data banks in hierarchical multimachine complexes from M-4030 (M-4030-1) and SM-3 (SM-4), simulation modelling of analog, digital and analog-digital processes, for scientific-technical and economic calculations, for teaching systems and for data processing in SANE, has been developed to expand the capabilities of OS and to deepen their problems orientation.

The second-unit software of the SM EVM has been developed on the basis of the first-unit software. The second-unit software is being developed in the following directions:

expansion of specialized program module bases;

development of OS generating systems for development of efficient OS of multimachine distributed computer complexes and networks and for highly productive and reliable function-sharing systems;

expansion of the set of programming languages by inclusion of KORAL, PASKAL', APL, PL/M and other languages;

expansion of the set of PPP, including those for management of relational data bases, text processing, control of production processes and so on;

development of SAPR and production of software and output of program documentation for mini- and microcomputers.

This system will be developed for joint design of computer complexes from the set of hardware and software as the nomenclature of equipment-realized program modules is expanded.

It may be noted in summarizing the brief consideration of the characteristics of the first-unit of the SM EVM that the SM-4P processor becomes the basic model of the processors in the family of small computers in the socialist countries, which is determined by configurational, technological and design capabilities of the SM-4, which provide:

the highest scales of output;

the most complete equipping with software (OS of nine types);

development of work to create special processors (SM-1600 for economic calculations, complex shop control systems, SM-1410 for scientific and technical calculations and programming automation), special processors for language and fast Fourier transforms and so on;

development of programs for remote processing, network structures and support of effective operating modes with YeS EVM;

development of compatible models of small and microcomputers at Minelektronprom [Ministry of the Electronics Industry] and Minradioprom [Ministry of the Radio Industry] (Elektronika-60, Elektronika 100/25, Nairi-4 and so on);

implementation of a broad program for development of problem-oriented complexes in all classes of ASU [Automated control system]: ASU TP [Automated production process control system], ASUP [Automated production control system], SANE, in monitoring and measuring systems and in the nonindustrial sphere.

Positive experience has been accumulated in the USSR by the Academy of Sciences and by a number of sectors of industry in compilation of coordination plans for complex automation of a number of subsectors based on models of the SM EVM. This ensures concentration of the efforts of the head organizations--developers and manufacturers of SM EVM and also of the sector systems organizations in solution of specific problems on development of POK and of pilot models of a broad range of control systems and reduces the deadlines for mass introduction of SM EVM. Because of this, it becomes possible to construct complex control systems based on equipment of the same type.

Large-series output of small computers determines the need to improve the process of control system development. A large number of highly qualified specialists would be required to introduce several thousand computers during the year with the existing traditional methods of ASU development at NII [Scientific research institute] when considerable human and time resources are required for development of a single system. Expansion of the spheres of application and complication of the problems being solved should be taken into account in this case, which in itself significantly increases the laboriousness of developing modern control systems. The principles of problem orientation of the SM EVM were developed and are being

applied to resolve this contradiction within the program for development of the SM EVM [5].

Main Trends of SM EVM DEvelopment

The complex program of investigations to develop and use the hardware and software of the SM EVM for 1981-1985 determines the main scientific research and experimental design work of the second-unit SM EVM carried out in the USSR and is an integral part of the work carried out by CEMA countries within the framework of the Intergovernmental Committee on Computer Equipment. The complex program provides for development of work in the following main trends: computer models, peripheral equipment, USO, a device for remote data processing systems and computer networks, software and standard computer complexes. At the same time, as already noted, all these integral parts of the complex program of the SM EVM solve the main problem--provision of the given level of problem orientation of hardware-software complexes of the SM EVM.

All models of the second-unit EVM are being developed on the basis of microprocessor sets of BIS and increased integration circuits, including those on 8- and 16-digit microprocessor sets based on n-MOP technology and also on microprocessor sections based on low-power TTLSh technology.

Models of this class are designed for constructing computer complexes of different configurations and can be used both in local systems (for example, in economic station based on the SM 1800, Figure 5) and in hierarchical and distributed control systems, including those in remote data processing networks. Computers, program-compatible with the SM-3 and SM-4 and designed for operation at the upper level of hierarchical control systems, have an adaptive configuration and, besides the main mode, can realize an operating mode which provides program compatibility with models of the YeS EVM and also if necessary with other computer systems.

The complex program envisions work on development of new models of peripheral equipment with solution of the following problems: an increase of recording density in magnetic carrier storage devices and the speed of data exchange and an increase of the productivity of impact printers and improvement of the ergonomic qualities of display devices.

Investigations of the new principles of developing peripheral equipment will make it possible during the 11th Five-Year Plan to develop optoelectronic information recording devices on a photothermal carrier, graphical and symbolic information input-output devices of fluid and laser type, display devices using laser displays and gas-discharge panels, medium-integration microcircuits based on TTLSh technology, memory BIS (static and dynamic OZU based on n-MOP and K-MOP technology), semipermanent storage devices based on TTLSh technology and programmed logic matrices based on TTLSh technology.

Computers of the following main classes have been proposed in the computer models.

1. Microcomputers designed for autonomous use, development of intellectual terminals, terminal stations, industrial controllers and so on. Output of microcomputers in the following design versions is provided with regard to systems applications, besides the main version: for use under especially complex operating conditions and for SANE (in designs of the CAMAC system).

2. A number of computers, program-compatible with the first-unit computer (SM-1--SM-4) and having higher technical and economic and operating indicators. It is suggested that special processors of different orientation (matrix, language, file and so on) be included in these models, which permits an increase of the speed of these models by one or two orders when executing the corresponding algorithms.

The program provides for development of a wide nomenclature of USO realized on integrated microcircuits, which are components of functional assemblies: operating amplifiers, operators, digital-analog converters and so on. The following main versions of systems applications of USO have been proposed: autonomous, distributed and mixed. Direct communications of the control facility with small or microcomputers is provided in the autonomous version of the USO and communication through an intermediate microcomputer is provided in the distributed version.

Development of the nomenclature of devices required to construct computer complexes, distributed control systems and computer networks in which communication between individual components of the system is accomplished through standard communications channels: telegraph, allocated and commutated telephone lines and physical lines, has been planned. Besides the traditional devices of data transmission equipment (APD, modems, MPD and so on), the program includes development of terminals and terminal stations which provide development of functionally complete remote data processing systems on a unified hardware-software base, beginning with multiterminal stations and ending with remote network processing complexes.

A significant increase in the level of the technology of programming systems and applied problems is provided in the field of software (PO), which is provided due to creation of developed instrument programming systems. The program provides for development of basic PO to construct highly productive multimachine and territorially distributed systems of increased reliability, software for construction of local and distributed collective-use information systems, test-diagnostic PO, instrument-production PO designed for preparation of programs, output of documentation and teaching of users and PPP for the more mass applications of the SM EVM.

The program encompasses assimilation and production of a number of standard multimachine configurations of computer complexes which may be the basis for design of specific user complexes. These complexes primarily include a number of two-machine complexes with different organization of computer interaction. A set of multiterminal complexes is designed for distributed data gathering and processing systems in industrial and nonindustrial fields. Complexes for use in distributed control systems, including in network remote data processing systems, occupy a special position. Besides general-purpose complexes, development of instrument complexes is provided for development and debugging of the PO of the SM EVM. Standard SM EVM complexes should be the basis for development of a large number of POK.

Implementation of the complex program will satisfy the most massive requirements of the national economy for use of SVT by expansion of the fields of application of SM EVM equipment and by increasing the efficiency of using this equipment. Achievement of the indicated goals will ensure solution of the following main programs of problems: expansion of the functional capabilities of SM EVM equipment, an increase of their technical and economic indicators and improvement of reliability and operating characteristics of SM EVM equipment, an increase in the level of programming technology, assimilation of mass production of SM EVM equipment and delivery of hardware, PO and service equipment in complete sets.

Fulfillment of the program will permit emergence to a leading level in all fields of application by the main user parameters: by functional completeness, productivity, complexing ability (with regard to development of programming equipment), reliability and viability.

BIBLIOGRAPHY

1. Naumov, B. N., "The International Small Computer System," PRIBORY I SISTEMY UPRAVLENIYA, No 6, 1977.

2. Rezanov, V. V. and V. M. Kostelyanskiy, "The SM-1 and SM-2 Control Computer Complexes," PRIBORY I SISTEMY UPRAVLENIYA, No 6, 1977

3. Dedov, Yu. A., M. A. Ostrovskiy, K. V. Peselev et al, "Malyye EVM i ikh primeneniye" [Small Computers and Their Application], edited by B. N. Naumov, Moscow, Statistika, 1980.

4. "Avtomatizirovannyye sistemy upravleniya na osnove POK" [Automated Control Systems Based on POK], edited by S. N. Khrushchev, TRUDY INEUM, No 71, 1978.

5. Morozov, A. A. and A. A. Stogniy, "Problems Orientation in ASU," UPRAVLYAYUSHCHIYE SISTEMY I MASHINY, No 3, 1978.

6. Naumov, B. N., and K. V. Peselev, "Malyye EVM v sfere upravleniya" [Small Computers in the Control Sphere], Moscow, Znaniya, 1979.

7. Filinov, Ye. N. and V. P. Semik, "The Software of SM-3 UVK," PRIBORY I SISTEMY UPRAVLENIYA, No 10, 1977.

8. Filin, A. V. and A. A. Solokhin, "Organizing the Processing of Interrupts in the COMMON BUS Systems Interface of the SM EVM," in "Tekhnicheskiye sredstva mini-EVM" [Minicomputer Hardware], No 61, Moscow, INEUM, 1977.

6521
CSO: 1863

UDC 681.325.5-181.4

PRESENT STATUS OF THE PROBLEM OF DESIGNING DATA FLOW COMPUTERS WITH NON-TRADITIONAL STRUCTURES AND ARCHITECTURE

Moscow IZMERENIYA, KONTROL', AVTOMATIZATSIYA in Russian No 1, 1981 pp 36-48

[Article by Academician I. V. Prangishvili of the Georgian SSR Academy of Sciences and G. G. Stetsyura, candidate of technical sciences]

[Text] In the course of the last 30 years the architecture of series-produced computers has remained fairly stable and has differed little from the architecture proposed by J. von Neumann. Modifications have been mainly evolutionary, and improvements have not led to radical changes in the structure and architecture of computers and programming methods. Efforts have been made to make new models compatible with preceding ones, as it was assumed that the enormous labor expenditures invested in software creation would force traditional computers and programming languages to be used for a long time.

However, work has been done recently on the creation of new programming methods and the development of new computers with a structure and architecture different from the traditional. The success of that work can radically change the situation in the area of computer technology. The efforts of researchers are being concentrated mainly in solving two problems. The first problem is the search for programming methods and computer structures which permit substantially simplifying and cheapening the creation and debugging of programs. It is known that about 80 percent of the cost of data processing systems is now expended on the writing and debugging of system and applied programs and, because of the steady cheapening of the hardware, the share of software cost in the total cost of systems steadily increases. In addition, with the gradual complication of control problems in real time the complexity of programs for them grows and program debugging becomes more and more laborious and costly.

In connection with that the designing of computers with a structure and architecture for which the programming of tasks requires no detailed description of the series of computations becomes extremely urgent. The programs obtained in that case are less unwieldy and simpler to debug.

The second problem is the search for a computer structure and architecture permitting the very effective use of microprocessors and other large integrated circuits. Here the main efforts have been directed toward the creation of multiprocessor and multicomputer systems which assure high speed and short response time.

In existing multimicroprocessor and multicomputer complexes problems are arising in organizing complex interconnection of asynchronously functioning parts of the complex, especially during the use of common resources. A large portion of the computer resources engaged by an operating system is expended on the organization of the joint work of separate parts. Therefore it is extremely urgent to search for new approaches to the creation of computers with a structure and architecture in which those expenditures are noticeably reduced through simpler and more natural organization of the interconnection of separate parts of the system.

One approach to which much attention is now being given is the development of computers executing flow models of calculation in which the flow of data arriving in the machine from below or obtained as intermediate results of computations controls the course of the computations. The computers are called data flow machines and traditional machines are called control flow machines.

The difference of data flow machines from ordinary computers consists in the following. Classical single-processor computer programs are organized linearly. Each instruction in the program has its own ordinal number (address) and is executed by following the instructions in the program. If that order must be changed the programmer must clearly indicate the address of the instruction which must be executed next. Indication of the order of execution will be substantially complicated in the transition to multiprocessor machines in which several instructions can be executed simultaneously. When computers are used to control objects the possibility of computation is determined by the presence of data reaching the machine from the controlled object at unpredictable moments of time. Due to complication of the task of putting computations in order it would be advisable to give up linear ordering of programs completely. Actually, the sole criterion of the possibility of making computations (with appropriate means of data processing) is the presence of data necessary for the computation. Therefore the computer structure can be more natural in which the instructions have not been put in order linearly, the values of data appearing in the course of the computations are directed into instructions where they can be used as the operand values and they are ready for the execution of instructions in the data processing equipment. Computations in data flow machines are organized in precisely that way. Rejection of the task of linear ordering in programs permits not stipulating many limitations unessential for the task, as a result of which program creation and verification ought to be substantially simplified.

A second important feature of data flow machines consists in the fact that data flow control is oriented toward application in multiprocessor and multicomputer systems. Finally, in systems where computations are controlled by data, it is possible in principle to achieve reactions to the appearance of new data more rapid and flexible than in traditional systems. It is precisely this which is required in most present-day real-time data processing systems.

Great interest is aroused in data flow machines by these features: simplification of program creation and verification, orientation toward multimicroprocessor and multimicrocomputer systems and effectivness of work in real time.

We will illustrate by an example the difference between the organization of computations in a data flow machine and in a traditional control flow machine [1]. Let there be the following fragment of a program in FORTRAN:

1. $P = X + Y$ 4. $S = R - Q$
2. $Q = P / Y$ 5. $T = R * P$
3. $R = X * P$ 6. $RESULT = S / T$

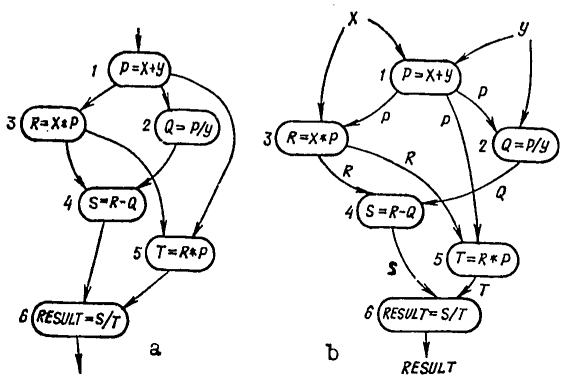


Figure 1. Graphs of a fragment of a computation program.
a -- initial b - converted

This program, presented in the form of graphs, is shown on Figure 1a. On an ordinary multiprocessor machine the sequence of execution of operations in that program would be different, for example: 1-3-5-2-4-6, 1-2-3-5-4-6, 1-2-3 parallel to 4 and 5 parallel to 6. Any operation at the apex of the graph can be executed asynchronously if all the operations from which arrows proceed toward that apex have been executed. In an ordinary multiprocessor system asynchronous computation with those limitations could be limited by using known synchronization methods, for example, "semaphores." However, those mechanisms are realized relatively slowly and are applied only on the macrolevel to control interaction of the processes. The subdivision into operations shown on Figure 1a is too "fine" for the use of "semaphores." A data flow machine usually is designed to work efficiently with execution in parallel on the level of individual instructions of the program. However, as will be shown below, data flow machine structures are known in which the processing in parallel is done on the macrolevel and is maintained by the hardware.

For a more graphic illustration of the process of computation on a data flow machine we convert the graph of the presented program to the form shown on Figure 1b, where the graph edges are marked with the names of the data controlling the course of the computation. The operations recorded at the apices can be executed asynchronously and independently of one another if all the data indicated on the inputs arrive on them.

The organization of computations in accordance with the graph of Figure 1b reminds one very much of the functioning of Petri networks [2-4], often used as the formal apparatus for description of parallel computation circuits. The network model is used in a number of programming languages for data flow machines [5-7].

In order that, in the presence of the necessary starting data, it would be possible to start executing the operations shown on Figure 1b, in the program for each operation there must be a logic design which detects the presence of all input data for that operation. Such a design is a particular case of a trigger function, the predicate, introduced in [8]. Whether or not the given data processing statement can be executed depends on its value. We also will call that design a trigger function.

Either the trigger functions must be indicated in the text of the program or each instruction must have the properties of a trigger function. During the physical realization of computations there is a mechanism which computes the values of the trigger functions of the program and selects the instructions ready for execution on hardware free at the moment. For that purpose, in particular, extremely developed means of operative information exchange are included in data flow machines.

Data flow machines in which the processor executes instructions only if the values of all its operands are ready for a specific operation are called data driven machines [7]. Often it is precisely they which are classed as machines of the data flow type. In data flow machines another method of data flow drive also is possible, in which the computation is made only when the result of a specific operation is required, that is, if there is a demand for computation. Such machines are called demand driven [9]. Data driven machines provide more rapid drive than demand driven machines, but the latter regulate the course of computations more precisely. Combinations of those two types of drive in a single machine are possible. Let us examine the generalized functional working diagram of data flow machines presented on Figure 2.

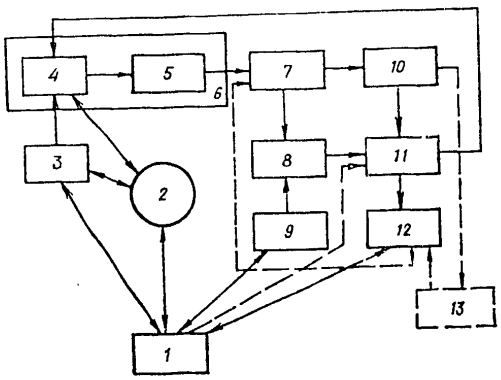


Figure 2. Generalized functional diagram of a data flow machine.

After the program and starting data have been loaded from the subsystem for control of mass stores and input/output equipment 1 under the control of a supervisor 2, and data with a complex structure into the data structures memory 9, the computation control system 6 starts to function. The latter consists of storage devices for incomplete data processing operators 4 and storage devices for operators ready for computation 5. Information on the appearance of data arrives at device 4 through the computation results distribution system 11 from the scalar data processing

device 8. The values of data from the external world also can be input through system 11 into device 4 (the broken line from 1 to 11). On the basis of information about the available data device 4 finds operators for which all the starting data are ready. The further work of device 4 depends on the method of computation control adopted in the data flow machine.

In the data driven method, operators having all the starting data are transferred to device 5. In the demand driven method, to transfer the operator to device 5 a request must arrive in device 4 for the execution of that operator. The task of device 5 is to transfer each operator present in it for execution in device 8 or into one of the group of devices 10. Either an arbitrary or a specific device from group 10 can be selected if it consists of devices of the same type or any device of a definite type (when there are devices of various types).

Each data processing device, having obtained the results of an operator's computation, directs them through system 11 into device 4 or into the computation results storage 12 if a definitive result has been obtained. Data can be transmitted from device 12 into subsystem 1 for output from the data flow machine. Information transmitted to device 4 is used in further computations in the system.

Let us dwell in greater detail on the form of the messages transmitted to data flow machines. They usually are called packages. A package of starting information transmitted to a processing unit contains the operator description or name, the starting data, limitations on the type of processing unit, the address or name of the operator-receivers of the result and auxiliary control information about the state of the system (the values of various flags). The package of results transmitted from the processing unit contains the value of the result of the operation and the address (name) of the operator-receiver of the result. Possible besides indications of the addresses of receivers of the result is the reporting of all units interested in obtaining a specific result (broadcasting), which enables the dynamically variable needs for the results of the computations to be better taken into consideration. Actually, let us assume that only as a result of computations can it be established which operand must be taken as the input for a certain operator. Since there was no such information at the moment the program was loaded in the data flow machine, the method of indicating the name of the receiver in the operator source is complicated extremely. In addition, the absence of the addresses of receivers in the packages can accelerate the exchange procedure.

The method of direct placement of the data to be processed in the package is considered redundant [10]. The method shown on Figure 2 by the broken line can be an alternative. Only the data addresses are indicated in the packages. The data processing unit, upon receiving any result, places it in the memory 12. If it is required for any processing unit to start the computation, it is returned first of all to unit 12 through arbitrage circuit 13, which resolves the conflicts when many devices are returned simultaneously. From unit 12 the required value is selected which is directed through coupling system 7 into the requesting unit.

In accordance with the schematic diagram presented on Figure 2, data flow machines are constructed which belong to the type of machines with many data flows and many instruction flows--MIMD. Data flow machines are usually constructed according to precisely that type, as in that case very great asynchronism of computations can be

achieved. However, data flow control can also be used in machines with many data flows and a single instruction flow--machines of the SIMD type [11].

The data flow machine [mashina upravlyayemaya potokom dannykh--MPD] scheme is generalized (see Figure 2). The functional diagrams of specific MPD's can be greatly different from it. Thus, whereas the diagram of the MPD-1 machine is very similar to that depicted on Figure 2 (MPD-1 - MPD-6 will be described below), in the MPD-2 microprocessor machine only some processors have such a structure. In the MPD-4, MPD-5 and MPD-6 machines the communication system represents a single whole. In far from every MPD are there devices for processing complex data structures. Distinctive features of those devices will be examined in the description of the MPD-1 machine.

The latest MPD developments are designed for single assignment languages [1,12,13]. According to that principle the "name-value" correspondence once established in the process of computation cannot be changed. If a new value must be given to a name, a new name with a corresponding value is introduced. We will illustrate the work of a single assignment MPD on a simple example.

Let the starting value X = 1 be given and the computation fragment contain the two operators:

$$1.\ X := X + 2$$
$$2.\ Y := 3X.$$

Let the two operators have the properties of a trigger function. Then each of them will be able to reveal the presence of the operand X = 1 necessary for work and start the computation. But if the operators work asynchronously and if special measures are not taken, the value of Y will be different, depending on whether operator 1 worked or not before operator 2. The situation is complicated still more if the operators are changed slightly:

$$1'.\ X := X + Y + 2$$
$$2'.\ Y := 3X + Z.$$

Here the operators 1' and 2' are executed in an unpredictable time after appearance of the values of Y and Z.

The principle of single assignment forbids the use of operators of the type of 1 and 1', that is, a new value cannot be assigned to an already created correspondence X = 1 (X = 3 for operator 1). Therefore instead of operator 1 it is necessary to write operator U: = X + 2, and instead of operator 1', operator U: = X + Y + 2. In that case ambiguity is eliminated.

In the presence of iterations in programs the "name-value" correspondence must be changed. For that purpose values of the types new X = X + 1, etc, are introduced, the new name new X is created which is real only for the given iteration, and only its value changes. Therefore outside the iterations there are no changes in the values of the data. Thus the single assignment principle is preserved even in programs with cycles [14].

The use of the single assignment principle is greatly complicated during the transition from scalars to completely organized data structures. In writing a program it is necessary to handle data organization very attentively, as the need to change a small portion of the structure can require copying the entire structure [13].

The advantages of the single assignment principle consist in the fact that with its application program writing and verification are simplified, complex references in programs are eliminated, data are used only locally, etc [1,13,14].

We will list the main characteristics distinguishing different types of MPD's:

1. Data drive is used.
2. Demand drive is used.
3. Computation is executed in parallel on the level of separate program instructions.
4. Computation is executed in parallel on the level of the group of instructions (subroutines).
5. The machine structure corresponds to the SIMD type.
6. The machine structure corresponds to the MIMD type.
7. A circular communication network or common bus is used.
8. A tree type of communication network is used.
9. Centralized commutators are used.
10. The single assignment principle is used.
11. There is a common memory in the system.
12. There are means for work with complex data structures.
13. Receiver addresses are used for distribution of computation results.
14. Broadcasting communications are used for the distribution of computation results.

Table 1 presents the results of comparison on the basis of those characteristics of six well-known data flow machines. They are conventionally called MPD-1 - MPD-6 in the article.

Table 1

A Условное название МПД	B Разработчик или тип МПД	C Признак 1	2	3	4	5	6	7	8	9	10	11	12	13	14
1 МПД1	2 Деннис	+	–	+	–	–	+	–	–	+	×	+	+	+	–
МПД2	LAU	+	–	+	+	–	+	–	–	+	+	–	–	+	–
МПД3	3 Румбах	+	–	+	+	–	+	–	–	+	+	+	+	+	–
МПД4	DDM1	+	–	–	+	–	+	–	+	–	+	–	+	+	–
МПД5	AMPS	–	+	–	+	–	+	–	+	–	+	–	+	+	–
МПД6	MAUD	+	–	–	+	–	+	+	–	–	+	+	–	+	+

A -- Type of MPD machine
B -- Developer or MPD type
C -- Sign
1 -- MPD-1, MPD-2, MPD-3, etc.
2 -- Dennis
3 -- Rumbaugh
+ -- presence of characteristic
x -- use possible
- -- absence of characteristic

Summing up the results of what has been said about MPD's, we can draw the following conclusions:

1. Many have proposed constructing MPD's with a very different structure. Whereas in some machines the execution in parallel is done on the level of a separate elementary instruction, in others it is done on the level of the subroutine. It is difficult to compare the effectiveness of different MPD's, and there are still no such data in the literature.

2. It has been noted [13] that, although since the moment of appearance of the first work on data flow machines (1962) many variations of those machines have been proposed, among them there is no sufficiently effective structure for universal application and especially for work with complex data structures.

3. In a data flow machine an extremely large role is played by the method of organizing communication between machine units. Primary attention should be given to questions regarding the organization of rapid commutation, announcement of the appearance of data and methods of controlling the course of computations. The data driven method can assure more rapid control than the demand driven method, but the latter permits more precise regulation of the course of the computation. Evidently it is advisable to combine the two methods in a single machine.

References in the instructions which indicate the instruction-receivers of the computation result can occupy a large volume and lead to a lengthy procedure of information exchange; in addition, it is difficult to use them in a dynamic determination of the receivers of results. Therefore, together with indication of the receiver's address, wide announcement of all potential users of the result of its appearance should also be used. Insufficient attention is still given to this matter.

In data flow machines it is preferential to have a method of organizing communications which permits readily changing the composition of the system equipment and fairly simply accomplishing communication both between adjacent and remote data processing equipment. Of interest in connection with that is the structure of communication of the ring type or the common bus, in which data are exchanged at a high velocity.

Among the methods of commutation control the application of which is useful in MPD's, one can point out different variants of decentralized priority control of data exchange [15], oriented toward work with a common bus and permitting distribution of commutation equipment over the system. Besides assuring a fairly high commutation rate the application of decentralized priority control in a number of cases permits using means of communication not only for the transmission of messages but also for making computations.

4. At the present time much work is being done on the organization of computations different from the organization proposed by von Neumann. That work is not reduced to just the design of data flow machines. Many of them are extremely interesting and the theoretical developments are immediately accompanied by proposals to create a corresponding machine. An example is the development by J. Backus of the principles of a functional programming language which facilitates the process of program writing, formal conversion and verification [16]. Following that appeared the

description of a machine (with the features of a data flow machine), capable of working with the Backus language [17].

The different directions of investigations of new data processing methods are not isolated ones, but influence one another. In such an unstable situation it should be expected that among MPD developments those which have a fairly flexible structure, one capable of rapidly adapting to new technical, algorithmic or linguistic developments, are the most viable.

5. In [13] it is noted that industry still displays little interest in MPD's. Besides novelty of approach and inadequacy of development of a number of important questions about their structure, this is explained by the fact that the MPD advantage in simplicity of program conversion and verification, assurance that computations are made in parallel, and the designing and use of program libraries. In addition, it is difficult for computer hardware producers to give up the accumulated mass of programs written traditionally. Nevertheless, interest in MPD will undoubtedly grow. This is connected above all with successes of integrated circuit production technology. Now the microcomputer, that is, a device which can work asynchronously and enter into complex interrelations with other components of the system, is becoming a separate module which can be used for the construction of intricate computer complexes. It is proposed to construct systems containing hundreds and even thousand of microcomputers. The time is arriving for control methods which maximally simplify for man the setting of the order for making computations in the system. Data flow seems very suitable for the implementation of those methods.

As examples of the construction of new untraditional computers we will examine six different MPD structures. They differ essentially from one another in the organization and method of computation. Yet the machines have important common features which are reflected in the generalized diagram (see Figure 2).

The MPD-1. The development and realization of MPD structures were first started in the computer laboratory of Massachusetts Institute of Technology by Professor J. B. Dennis [18]. Initially envisaged was the organization of MPD's not designed for work with large masses of complexly organized information and intended for the digital processing of signals [19,20]. Devices for work with information with a complex structure were introduced into the machine [21]. Figure 3 shows the organization of such an MPD.

In the instruction storage one instruction is contained in each cell. Through the distributor into the instruction storage cells arrive operands and signals of confirmation, which will be discussed below. If a cell containing an instruction receives all the operands and confirmation signals, it goes into an excited state and its contents are referred through the selector into the processors section to the appropriate processor for execution. All the information necessary for processing--operands, the operation code, the memory location of the instructions and the receivers of the results, is formed in the form of a package which the selector also transmits. The selector can simultaneously connect many instruction storage cells with processors. The instructions processing scalar values are directed into scalar processors, and instructions on the processing of complex data structures into a data structure processor. Each result of processing is formulated as a package of the result containing, besides the computed values, the addresses of receivers of

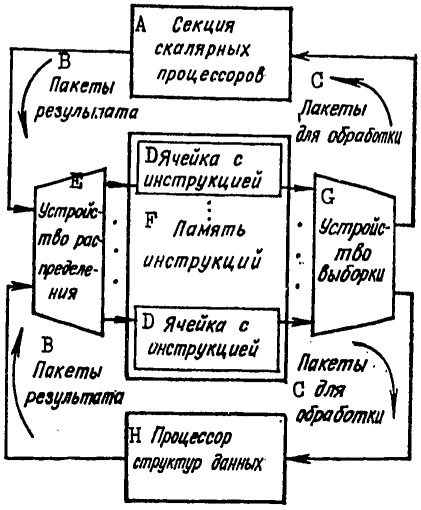


Figure 3. Organization of the MPD-1 machine

A -- Scalar processors section
B -- Packages of result
C -- Packages for processing
D -- Cell containing an instruction
E -- Distributor
F -- Instruction storage
G -- Selector
H -- Data structures processor

the result. Result packages arrive at the distributor, which refers them to corresponding instruction memory cells. In such a system blocking and even a dead-end can arise if the instruction is multiply transmitted for execution without waiting until the preceding result of processing that instruction is transmitted to all receivers. To eliminate such a situation signals confirming reception of the result from the instruction-receiver to the instruction-source of the result.

The data structure processor [protsessor struktur dannykh--PSD], a diagram of which is presented on Figure 4, received packages for processing with the instructions "create," "select" and "add." The PSD consists of a package memory system [sistema pamyati paketov-SPP] and three units forming the PSD controller: an interpreter, a queue control unit and a transmitter.

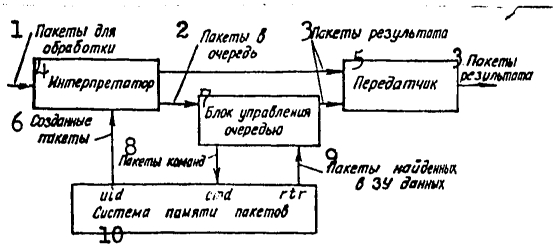


Figure 4. Diagram of an MPD data structure processor

1 -- Packages for processing
2 -- Packages in queue
3 -- Packages of result
4 -- Interpreter
5 -- Transmitter
6 -- Created packages
7 -- Queue control unit
8 -- Instruction packages
9 -- Packages of data found in stores
10 -- Package memory system

The SPP is intended for storage of the description and values of data structures. Each data structure value contained in the SPP has its own name, representing the value of the structure in all units outside the PSD. Inside the SPP the data structure values (in [21] only masses of data were examined) are represented by an element of the type (i, (c_m, m, c_n), r), where i is the data structure name; c_m, ..., c_n is either all real numbers or all the names of structure components. If the k-th component has not been determined the corresponding c_k has the value nil; r is the counter of references, indicating the number of references per element. When all references disappear, the element can be with drawn from the SPP.

The SPP content is processed by means of the following instructions;

--the instruction "store." A package containing this instruction, having the form <STO, i, k, c> enters the CPP in the port cmd. According to this instruction the value c is assigned to the component c_k of the element with that name, c_k = = c, and its reference counter is established as equal to unity;

--the instruction "retrieve." The package corresponding to it has the form < RTR, t, k> and enters the CPP in the port cmd. If in the SPP there exists the element (i, (c_m, ..., c_n), r) with m < k < n, then the retrieval package <i, k, c_k> is referred to the port rtr;

--the instruction UP. Upon arrival on the cmd input of the package < UP, i > the content of the reference counter of element i is increased by unity;

--the instruction DOWN. Upon arrival on the cmd input of the package < DWN, i > the content of the reference counter of element i is reduced by unity. If in that case r ebecomes equal to zero the element is eleminated from the SPP and its name is put on the list of accessible names;

--the instruction UID (unique identifier generation). The package < i > is referred to the port uid and the name i is eliminated from the set of accessible names.

The interpreter translates instructions "identify," "select" and "add" to programs of the above-indicated five instructions. The transmitter generates packages of results and confirmation signals for reference to instruction storage cells. The queue control unit, the basis of the controller, watches each "selection" operation started and not completed. For correct data processing in the MPD it is required that the retrieved values be issued from the data structure processor in the order indicated in the operation "select," in spite of possible change of the order during retrieval in the SPP (the retrieval can be carried out in parallel in the SPP). The queue control unit eliminates possible disruptions in the order of information issuance.

In an article [21] the work of the MPD is illustrated by solution of the task of weather prediction. That task is well carried out in parallel. The GISS model, in which the state of the atmosphere is characterized by four variables and partial derivatives in spherical coordinates, is used for global prediction of weather for one day. The computations are made on a three-dimensional network which breaks

the atmosphere down into 9 intervals by height, 45 intervals by longitude and 72 intervals by latitude with the time step t = 5 min. On an IBM 360/95 computer such a forecast is made in an hour. For long-range forecasts the complexity of computations increases by two orders of magnitude.

To solve that task an MPD must process a column of cells of a network with the given longitude and latitude in 40 microseconds. The computation program occupies 13,000 instruction memory cells and the computation of a new value for all cells of a network with an identical longitude and latitude requires about 7000 packages for processing. Of them there are 2700 packages of multiplications and divisions, 2900 of additions and subtractions, 900 operations with data structures and 700 packages with auxiliary operations. To complete those operations in 40 microseconds the group of scalar processors must process packages with a frequency of 150 MHz, and the data structure processor with a frequency of 25 MHz. Commutation must proceed with a frequency of 175 MHz. The queue control unit memory must be large, as at a frequency of data arrival of 25 MHz a delay of even 1 millisecond in retrieval requires storage of 25,000 retrieved values.

At the present time the question of physical organization of the controller and SPP in the data structure processor has not been solved.

The MPD-2. In the CERT Institute in France the data-driven LAU machine is being developed [22,23]. The principle of single assignment is used in the LAU machine. The following types of execution in parallel are possible in it:

-- on the program level (parallel multiprogramming);
-- on the level of tasks of a single program;
-- on the level of task instructions;
-- within the limits of an instruction.

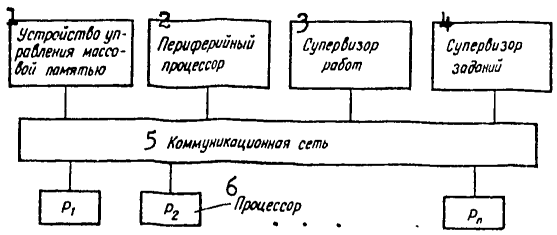


Figure 5. Structural diagram of the LAU machine

1 -- Mass memory control unit
2 -- Peripheral processor
3 -- Work supervisor
4 -- Tasks supervisor
5 -- Communication network
6 -- Processor

Figure 5 presents a structural diagram of the machine. The work supervisor controls multiprogramming in the system; the task supervisor controls the processing of parts of the task; the peripheral processor accomplishes external connections of the machine; the communication network accomplishes all communications inside the machine; the processors P_1, P_2, ..., P_n accomplish the data processing.

The machine instructions have the format:

KOP	Address of result	Address of operand 1	Address of operand 2	C_0, C_1, C_2

The left part of the instruction is stored in the local memory subsystem of the processor P (see Figure 6), the right--the flags C_0, C_1 and C_2-- is stored in the instruction control memory ICM. Any instruction can be executed on any processor of the machine. The number of processors can be increased. Bit C_1 indicates the readiness of operand 1, bit C_2 readiness of operand 2, and bit C_0 the indirect influence of other instructions on the data. The instruction set includes instructions which can, by changing the value of C_0, allow or forbid the execution of the other instructions. Instructions can be executed if $C_0 = C_1 = C_2 = 1$.

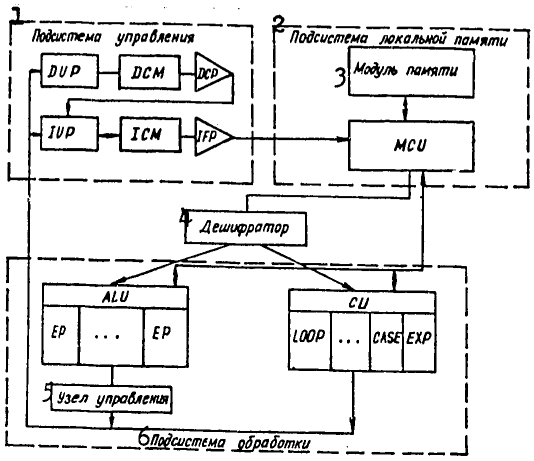


Figure 6. Block diagram of the LAU machine processor

1 -- Control subsystem
2 -- Local storage subsystem
3 -- Storage module
4 -- Decoder
5 -- Control unit
6 -- Processing subsystem

The data have the format:

Value L/R Connection 1 L/R Connection 2 C_d

Bit C_d is stored in the data control memory DCM and indicates whether the data values were calculated or not. Connections I and 2 indicate the address of the instruction using the operand and the L/R bits indicate which operand is used, the left or the right.

The arithmetic instructions of the machine compute the result and report on that to the instructions noted in the indicator "communication." Controlling instructions relate to the upper language. They react to flags, for example, clear the sets C_0 and change and verify the sets C_1 and C_2.

The local memory subsystem consists of several memory modules controlled by the memory control unit MCU, which controls all references to memory, using a priority mechanism to resolve conflicts.

The processing subsystem executes arithmetic and control instructions. It consists of two groups of processors on which instructions arrive from the decoder. The group ALU of arithmetic processors contains several elementary processors EP of

the same kind working asynchronously. The instructions can be executed on any free EP. The group of control processors executes complex control instructions--cycle control (LOOP), control of execution in parallel (CASE, EXP), etc [23].

The control subsystem accomplished data control. It contains two control memories: instruction ICM and data DCM. The ICM contains flags of instructions C_0, C_1 and C_2. The control processor IUP changes the ICM content on the basis of information transmitted to it from the processing subsystem. The selection processor IFP finds in the ICM instructions with $C_0 = C_1 = C_2 = 1$, that is, ready for execution. The DCM memory contains flags C_d of data. The control processor DUP changes the content of the DCM on the basis of information obtained from the processing subsystem. The verification processor DCP observes the readiness of operands and informs the IUP. Thus the processor P is the principal untraditional unit of the machine.

Analysis of various programs has shown that the task can be transmitted for execution in parallel in 10-12 processors on the average. Starting from that, a machine will contain on each DCM module an ICM with a 50 nanosecond cycle, 1 DCP module, 4 local memory modules with a 500 nanosecond cycle, an MSU module with a 100 nanosecond cycle; the processing subsystem consists of 12 elementary processors EP and two control processors CU; the typical addition time is 100 nanoseconds.

More than 50 programs were simulated. Some results of the simulation are presented in Table 2. For comparison, the time required for running those programs on an IRIS 80 computer is presented. In capacity it is similar to the IBM 370/145.

Table 2

A Программа	B Время выполнения программы, мс	
	LAU	IRIS 80
1 Произведение полиномов	1,402	3,6
2 Произведение матриц	0,546	1,6
3 Вычисление полинома Лагранжа	1,596	12,6
4 Параллельная сортировка	3,052	4,8
5 Начисление зарплаты	1,396	2,4

A -- Program
B -- Operation time, ms
1 -- Polynomial production
2 -- Matrix production
3 -- Lagrangian polynomial computation
4 -- Parallel sorting
5 -- Counting of payments

The MPD-3. This computer has not been realized physically but has only been developed within the framework of the MAC program at M. I. T. [24]. Information is exchanged between data processing devices by means of packages. A package contains data and an indication of its type. A package is closed, that is, does not contain addresses of storage cells, references to other packages, etc. Demand drive is accomplished in the machine. The operator is ready to be executed

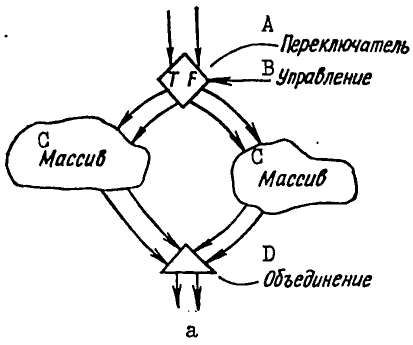
when packages have arrived on all the inputs. Execution of the operator depends only on the local information, and there are no global variables or side effects. The operators have no internal memory between computations.

The following control operators are used in the machine.

A switch. It has N information inputs, a dual control input and 2N outputs. One set or another set of N outputs is selected, depending on the value of the control output.

A union which unites inputs in a common OR output.

Provisional branching, formed by connection of the switch and a union in accordance with the diagram shown on Figure 7a.



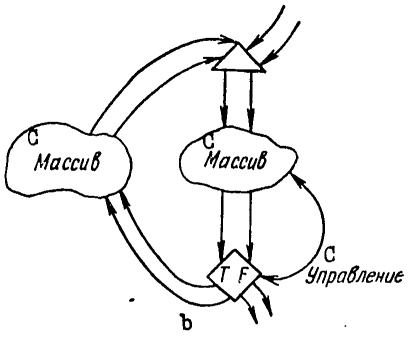


Figure 7. Diagram of the formation of control operators in the MPD-3 machine.

A -- Switch
B -- Control
C -- Array
D -- Union

A cycle. Figure 7b presents a diagram of the organization of this operator.

The program is constructed recursively. It consists of an operator, either an acyclic graph of programs or a provisional branching into unconnected programs or a cycle containing programs. The procedures can be copied and transmitted as

values. Applications for the procedure arrive from the operator Apply, to which the argument must also be given. Apply selects information from its inputs, arranges the argument on the procedure input and enables it to work within the operator Apply. When the procedure completes the work it is annihilated and Apply issues the result of the computation in the form of its output package. Also possible is an equivalent mode in which the procedure is performed not within but outside of the operator Apply, and after completion of the work the procedure is returned to Apply. The procedure can contain Apply operators within itself, that is, a tree-like inclusion of procedures is allowable.

The MPD-3 machine, a structural diagram of which is shown on Figure 8, consists of asynchronous modules. The activation processor AP causes and accomplishes separate

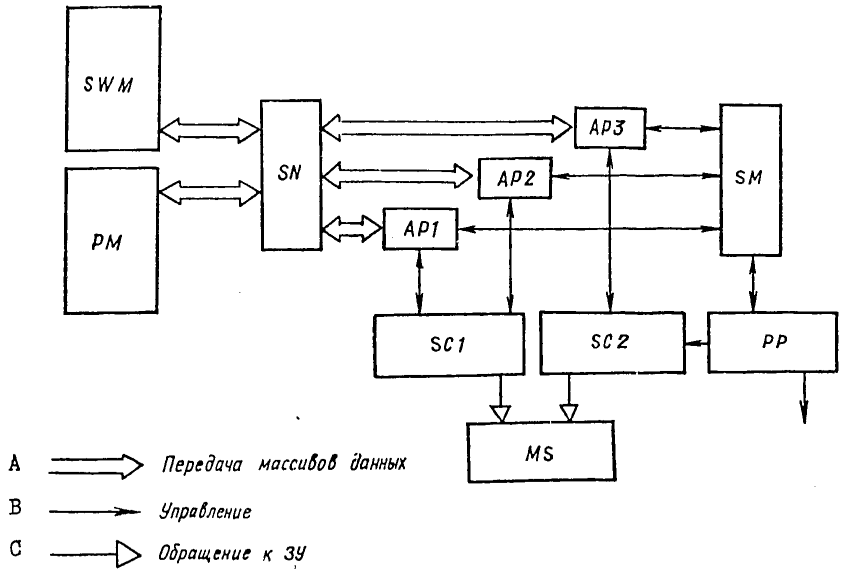


Figure 8. Structural diagram of the MPD-3 machine.

A -- Data arrays transmission
B -- Control
C -- Reference to memory

procedure activation. There are several AP's (which are shown on the figure), the planning module SM coordinates the work of processors and distributes activation among them. The memory of structures MS contains data structures in a quantity sufficient for loading the AP. The structure controllers operate with data structures for the AP. The program memory PM contains procedures which can be summoned from it. The exchange memory SWM contains activations of procedures which are temporarily inactivated. By means of the commutator SN the transmission of activations of procedures among SWM, PM and AP is accomplished. The peripheral processor PP serves for the organization of computer communication with the external world.

All the modules work asynchronously and interact, sending messages over a directed asynchronous channel connecting the pairs of modules. The working principle of the machine is very simple, as the behavior of the module is determined by the series of messages it transmits. Let us examine in greater detail the work of the processor of activation of that machine.

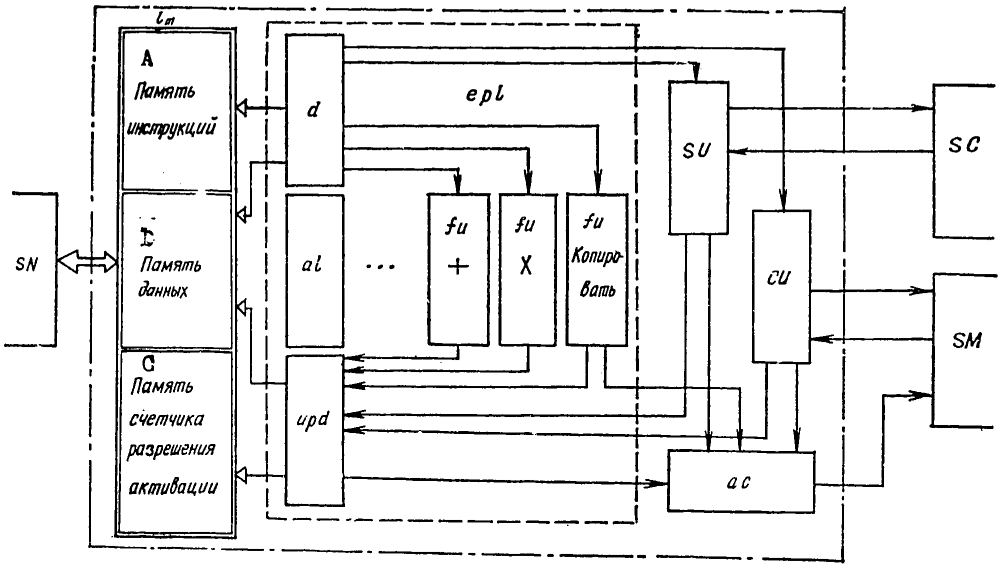


Figure 9. Schematic diagram of the MPD-3 machine.

A -- Instruction memory
B -- Data memory
C -- Activation resolution counter memory
D -- Copy

The activation processor AP (Figure 9) performs the program computation properly speaking. It contains the local memory lm, means of processing epl and an activation counter ac.

The local memory lm contains instructions, data and a counter of activation allowance. Each operator is coded as one instruction. It contains an operation code, operand addresses and a result, and addresses of instructions and users of the result. The resolution counter indicates the number of lacking operands of the instruction corresponding to it.

The means of processing contain an activity unit al, a decoder d, several functional units fu and a control upd. All the units form a closed main line of data processing in which several instructions can be executed simultaneously. The al unit contains the addresses of instructions ready for execution which await the possibility of execution. The decoders for the address from the al unit, selecting

operand values and instructions from the memory, forms an instruction package with the format

KOP	Operand 1	Operand 2	Address of result	Address of next instruction

That package is sent by the decoder to the corresponding fu unit for execution. The units are connected in parallel and work asynchronously and independently. The result of the computation is formed as a package with the format

Result	Address of result	Address of instruction-user of result

The upd unit receives packages from all processing devices and accomplishes arbitrage of them, places the result in the local memory lm and reduces the value of the corresponding counter. If the value of the counter becomes equal to zero the upd unit sets the counter in the starting position and places the address of that instruction in unit al to await execution. Each AP contains two functional units which have an output from the processor: a request unit CU which sends requests for procedure to the planning modulus SM and the structures unit SU, which sends requests for structure processing in the structure controller.

The activity counter is necessary for work of the planning module. It counts the number of active data processing devices in the AP.

Since the local memory has a small volume, large data structures are stored in the structural memory MS. Although in accordance with a general principle the data need not be used jointly, structures in the MS are represented as acyclic graphs in which the common substructures are sometimes used jointly. However, the SC controllers represent such structures as completely separate for the AP.

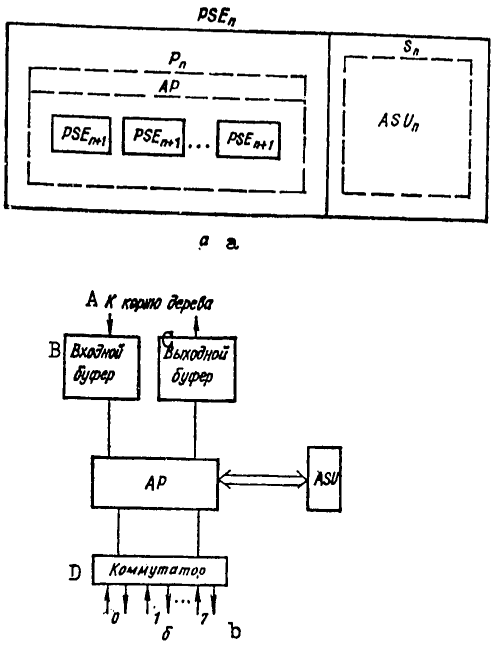
The planning module also creates, maintains and annihilates the activation of procedures and distributes them on processors for execution. The form of programs for that MPD was described in [9] and it was shown that they are correctly run. The absence of dead ends and runnings is guaranteed.

The user can communicate with the computer in a high-level language (ALGOL 60, for example), which is translated automatically into the required form [25]. However, in the work it is noted that it is advisable to use new languages oriented toward the principle of single assignment.

The MPD-4. The Burroughs Company has constructed a model of the DDM-1 machine. Further research is being done at Utah State University [26, 27].

The main computation unit of that machine is the "processor storage element" PSE, the structure of which is shown on Figure 10a. The PSE consists of the processor P and the local storage S. Any PSE can run any program of the machine if the storage volume S of the given PSE is sufficient. If the storage volume is inadequate the PSE can occupy storage in other PSE's, but that sharply diminishes the capacity of the machine. The machine is constructed of PSE's recursively:

$$\langle PSE_n \rangle ::= \langle P_n \rangle \langle S_n \rangle$$
$$\langle S_n \rangle ::= \langle ASU_n \rangle$$
$$\langle P_n \rangle ::= \langle AP_n \rangle | \langle AP_n \rangle \langle PSE\ GROUP_{n+1} \rangle$$
$$\langle PSE\ GROUP_{n+1} \rangle ::= \langle PSE_{n+1} \rangle | \langle PSE_{n+1} \rangle \langle PSE\ GROUP_{n+1} \rangle ,$$



A -- to root of tree
B -- Input buffer
C -- Output buffer
D -- Commutator

Figure 10. Schematic diagram of the "processor-memory" unit of the DDM-1 machine
a -- structure b -- tree-like organization

where the subscript designates the recursion level to which the module belongs; AP and ASU are the atomic processor and storage unit respectively, the structures of which are not broken down further, that is, do not contain other PSE's. The number of processors forming the PSE GROUP in the machine is fixed at eight. Thus the PSE has a tree-like organization (Figure 10b).

Commutators 1-8 are used to accomplish connections between adjacent levels in the PSE. Messages proceeding along the tree from the bottom up are taken on the basis of arbitrage. The data transmission mode is asynchronous, with conformation from the receivers. During information exchanges queues can form and, in order not to overload the processors, buffers are established on the inputs and outputs of the

latter. All the connections, with the exception of the connection AP-ASU, are accomplished on six lines (2 lines for the request and confirmation of connection and 4 for symbol transmission).

The character of control of the work of the machine and the variable length of the successively transmitted messages present high requirements for memory organization. Therefore all the internal functions with respect to memory control are accomplished within the storage unit ASU. It can accomplish independently such instructions as "pass," "insert," "read," "write," "erase," etc. The free memory is distributed automatically. In the DDM-1 machine the ASU consists of a ZUPV with a capacity of 4 K four-digit words. The input and output buffers have capacities of 1 K characters each.

The tasks solved on the machine are presented in the form of a data driven network DDN. The DDN has much in common with other similar network representations [28, 29]. The principal differences consist in the fact that, firstly, the package in the DDN carries rather capacious information and the execution in parallel occurs not on the level of individual operations but on the level of individual subroutines. The DDN also reminds one of a Petri network as regards the logic of the work.

Presented on Figure 11 is a DDN network fragment. Data are processed at the network units. If there is at least one package on all the inputs of the network unit the unit is "ignited," performs the computation and sends the result to other units connected with it. When that result has been received by all users, the work of the unit on the given computation is completed.

When the package containing the DDN program arrives at the PSE of any level, that unit performs one of the two actions:

1) if the PSE has a substructure and the tasks are broken down into simpler ones, one attempts to distribute the task to a PSE of a lower level for execution;

2) if those conditions are not observed, the PSE performs the computation independently.

The principal shortcoming of the DDM-1 machine is its fixed structure, which makes the redistribution of resources in the system difficult; the exclusion of all PSE's of lower levels during failure in PSE's of the upper level is a result of simplicity of the commutation system.

The MPD-5. The DDM-1 machine developed here has influenced the plan of the AMPS (Applicative Multi Processing System) machine developed at Utah State University. At present it is not proposed to realize the machine physically [30]. The AMPS has to represent a set of processors (about 1000) combined on the basis of weak couplings. Weak connection assumes parallelism on the level of separate programs, and not instructions. In contrast with many other machines, the AMPS is oriented toward the execution of dynamic refinements and mainly symbolic data processing. It is intended for such languages as LISP; the FGL language (Flow-Graph LISP or Function Graph Language) created for it is a LISP dialect.

The machine (see Figure 12) is organized in the form of a tree with two layers of units. The suspended apices of the tree contain processors and a memory. The

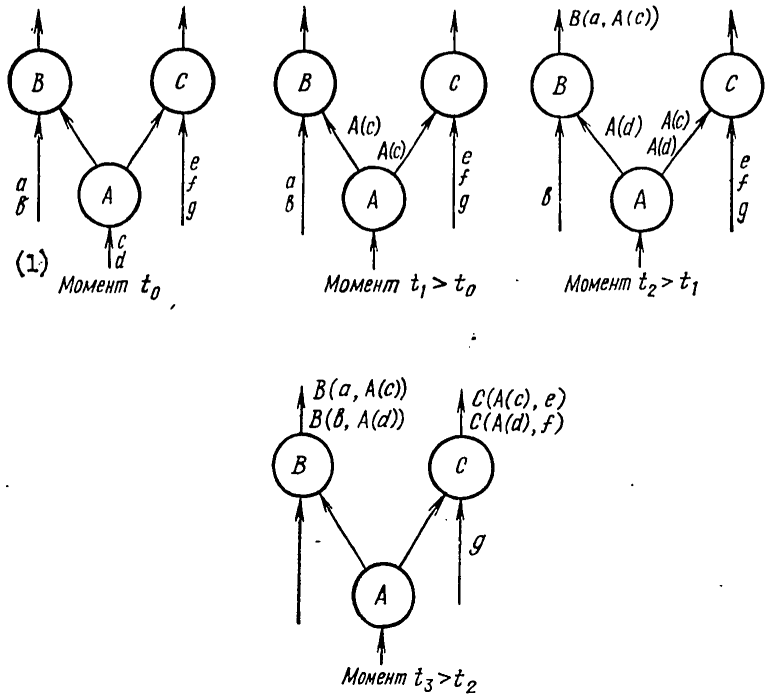


Figure 11. Fragment of DDN network:
A,B,C -- operators
a,b,c,d,e,f,g -- data
1 -- Moment t_0, etc

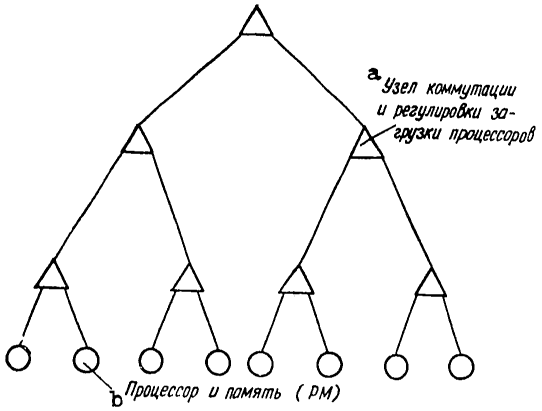


Figure 12. AMPS machine organization.

a -- Unit for commutation and regulation of the loading of processors

b -- Processor and memory (PM)

internal units contain commutators and devices for regulating the loading of processors. Shown on the figure is a double tree, but it is assumed that when realized it can be eight-dimensional or even more complex.

The processing device PM, arranged in the suspended apex, in its complexity is reminiscent of an ordinary microcomputer, but differs essentially in structure. The PM can accomplish data processing properly speaking, the reception of input operands and the sending of results over the tree, select information from other units of the tree and distribute the memory during processing of the instruction "invoke."

The instruction "invoke" creates a task which, depending on the loadability of the computer modules, can be executed on the given or another PM. The memory in the computer is distributed among the PM's by units with a capacity of 64 K words. Each PM can be referred by the commutation network to the memory of other PM's. The entire memory has a common field of addresses which are generated not on the level of translation but dynamically. Reference to the external memory and other peripherals is accomplished through specialized processors situated in the suspended apices of the tree.

The commutation network controls exchange in the computer. For exchange between suspended apices a message must be transmitted over the tree to the nearest apex from which release to the reserves is possible.

Information is exchanged on the basis of demand drive and after the task has been started it can actively request data. The information flow passing through the commutation network can be broken down into a flow of tasks (a flow of instructions "invoke"), a flow of operands (separate words of data), a flow of data units and a flow of requests for operands and units. The information is formulated as packages, each of which is provided with the address of the receiver and is transmitted in a mode of package commutation.

The tasks for this machine are presented in the form of a network which reminds one of a DDN network. Information is processed in the apices of the network provided that the starting data have been obtained: the results are sent over the branches if there is a request for them.

The instructions which comprise the program executed on a computer contains an operations code, local addresses of the arguments and indicators (local addresses) of instructions and users of the result. Information is stored and transmitted in the form of data units and instruction units. Local addresses are addresses within a unit. In the instructions global addresses can be used to establish connections between units, and also references to a unit in the instruction "invoke." "Invoke" accomplishes separation of memory for the data unit and copying of the unit into the separated memory and organizes connections between the apices of the task graph.

To control work in the AMPS machine a task list TL is used, one consisting of two lists with which one can work independently: a task list TLC containing addresses of instructions requesting computation and a results list RL which contains the addresses of instructions which have computed the value of the result. The instructions "invoke" are distributed among the PM's of the communications network

with consideration of the loadability of the PM. Each PM has its own list of instructions "invoke" which is a part of the TL. The communications network distributes the instructions "invoke" in accordance with those lists.

The task execution mechanism is as follows. First the address of the instruction which is to compute the desired result is placed in the TL. Then the instruction and the arguments indicated in it are extracted in accordance with that address. If all the arguments are ready, the computation is made. If some arguments are not ready, on them an application is made to the TL with indication of the address of the user, the result, a reference to the obtained result is placed in the CP, and the instructions awaiting that result inform about its being obtained. During realization of that mechanism the PM can require a word not in its memory. In that case exchange through the communications network is required, and the PM possibly has to await the result for a long time. For the PM not to stand, a buffer is provided for the storage of several tasks. If one task awaits delivery of information through the network, the other can be executed.

Load is distributed among PM's as follows. Each communications network unit strives to maintain the quantity of loading of subsequent network units connected to it in the given interval [L,U]. The lower L and upper U of the interval boundary are determined by the memory volume occupied by the computation. If the loading of one of the units is greater than U and of the other is smaller than L, then the corresponding commutation unit attempts to transfer the tasks from the list of instructions "invoke" of any overloaded unit to the corresponding list of the underloaded. If the overloading is not eliminated, a request is made higher on the tree in order to remove that task from the given subtree.

The MPD-6. The MAUD machine (Machine d'Assignation Unique Dynamique--Dynamic Unique Assignment Machine) is being developed at Lille University in France. The authors are trying to create a data driven machine in which a complex commutation system will not be required [31].

In MAUD the set of instructions, together with the set of objects (data) which those instructions use, are united in a unit. The program consists of a group of units. Exchange of objects between units is required for work of the program. Exchange is accomplished according to the principle of single assignment. The communication name is assigned for exchange of an object. That name is unknown inside the unit, where the internal names of objects are used. The unit consists of a coupling part connecting the input region (the i-region) and the output region (the o-region); an internal part consisting of processing operators and objects used by the operators; an interface part consisting of a list of correspondences of external and internal names of input objects and the same list for output objects (results). The unit works (issues output data) when there are all the necessary input objects.

Figure 13 shows a schematic diagram of work with the units. Region X contains units ready for execution which are used for work of the processor P. The results of work with the processors are placed in the o-region of S output objects. Units which temporarily cannot be accomplished and await the arrival of input data arrive from the processor P in region A of the waiting units. The control unit CB monitors regions S and A and upon arrival of the corresponding data directs units from region A into region X. If working units are required to form and put something

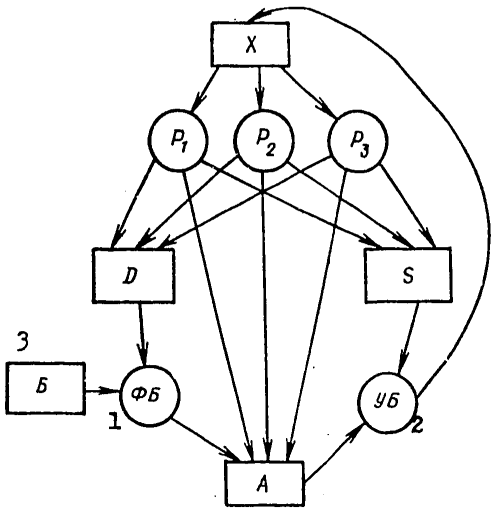


Figure 13. Schematic diagram of work with program units in the MAUD machine

1 -- unit former UF
2 -- control unit CB
3 -- library B

new to work, they send the request to the region of unit requests D. Upon request the unit former UF selects the unit code for the library B and directs it into region A. Thus the units do not interact directly with one another.

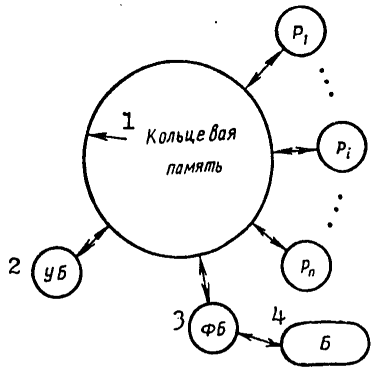


Figure 14. Structural circuit of the Maud machine

1 -- Circular memory
2 -- Control unit
3 -- Unit former
4 -- Library

Physically, the MAUD machine is realized in the form of a circular memory to which the processing and control processors are connected (see Figure 14). The memory consists in a shift register with ports for connection of the processors. The units circulate in the memory in the form of packages set in zones of equal length. Each zone has a starting marker and a flag which states the zone contents. The processors can record packages in a free zone and also liberate it after use of the information addressed by it. Table 3 shows possible effects of processors with the zone.

Table 3

Type of processor	Operation performed	Zone flag before operation	Zone flag after operation
Processing processor	Reading	Unit	Empty
	Recording	Empty	Application for unit
	"	"	Waiting unit
	"	"	Output unit
Control processor	Reading	Waiting unit	Empty
	"	Output object	"
	Recording	Empty	Unit for execution
Unit former	Reading	Application for unit	Empty
	Recording	Empty	Waiting unit

Thus all the regions indicated on Figure 13 are stored in the circular memory, and the processors interact only through that memory.

The system is in the stage of manufacture at the present time. It is considered that it will be advisable to introduce some control that will lead to more rational use of the circular memory.

BIBLIOGRAPHY

1. Ackerman, W. B. Data flow languages. AFIPS, 1979, pp 1087-1095.

2. Peterson, J. L. Petri nets. COMPUTING SURVEYS, 1977, Vol 9, No 3, pp 223-252.

3. Peterson, J. L., and Bredt, T. N. A comparison of models of parallel computation. INFORMATION PROCESSING, 1974, pp 465-470.

4. Kotov, V. Ye. Algebra of regular networks. Preprint, Institute of Mathematics, Siberian Department, USSR Academy of Sciences, Novosibirsk, 1978, 175 pages.

5. Davis, A. L. The architecture and system method of DDM1: a recursively structured data driven machine. SIGARCH NEWSLETTER, 1978, No 6 (&), pp 210-215.

6. Dennis, J. B. First version of data flow procedure language. LECTURE NOTES IN COMPUTER SCIENCE, 1974, Vol 19, pp 362-376.

7. Dennis, J. B. A language design for structured concurrency. LECTURE NOTES IN COMPUTER SCIENCE, 1977, Vol 54, pp 231-242.

8. Kotov, V. Ye., and Narin'yani, A. S. Asynchronous computational processes for a common memory. KIBERNETIKA, 1966, No 3, pp 64-71.

9. Rumbaugh, J. E. A parallel asynchronous computer architecture for data flow programs. MIT Project MAC, 1975, TR-156.

10. Farrell, E. P., Chani, N., and Treleaven, P. C. A concurrent computer architecture and ring based implementation. SIGARCH NEWSLETTER, 1979, Vol 7, No 6, pp 1-7.

11. Zatuliveter, Yu. S., and Medvedev, I. L. Group parallelism and dual realization of parallel computations. VOPROSY KIBERNETIKI, 1978, No 43, pp 44-64.

12. Tesler, L. G., and Enea, H. J. A language design for concurrent processes. AFIPS, 1968, Vol 32, pp 403-408.

13. Organick, E. I. New directions in computer systems architecture. EUROMICRO JOURNAL, 1979, Vol 5, No 4, pp 190-202.

14. Ashcroft, E. A., and Wadge, W. W. Lucid, a non-procedural language with iteration. COM ACM, 1977, Vol 20, No 7, pp 519-526.

15. Prangishvili, I. V., and Stetsyura, G. G. Macroprocessor systems. Moscow, Nauka, 1980, 236 pages.

16. Backus, J. Can programming be liberated from the von Neumann style? COM ACM, 1978, Vol 21, No 8, pp 613-641.

17. Mago, G. A. A network of microprocessors to execute reduction languages. INTERNATIONAL JOURNAL OF COMPUTER AND INFORMATION SCIENCES, Part 1, 1979, Vol 8, No 5, pp 349-385; Part 2 ibid, 1979, Vol 8, No 6, pp 435-471.

18. Workshop on data flow computer and program organization, transcripted by D. P. Misunas. COMPUTER ARCHITECTURE NEWS, 1977, Vol 6, No 4, pp 6-22.

19. Dennis, J. B., and Misunas, D. P. A preliminary data flow architecture for a basic data flow processor. Proceedings of the Second Symposium on Computer Architecture, 1975, pp 126-132.

20. U.S. Patent No 3962706. Data processing apparatus for highly parallel execution of stored programs. J. B. Dennis and D. P. Misunas. Published 8 Jun 76.

21. Dennis, J. B., and Wang, K. K-S. Application of data flow computation to the weather problem in High-speed computer and algorithm organization, 1977, pp 143-157.

22. Sure, J. C., et al. LAU System: a parallel data-driven software-hardware system based on single assignment. M., Fielmeier (ed), Parallel computers-Parallel mathematics, 1977, pp 347-351.

23. Come, D., Syre, J. C., et al. The LAU parallel system: software definition and implementation through a multimicroprocessor architecture. Euromicro Symposium, Preprint, 1976, pp 109-117.

24. Rumbaugh, J. A data flow multiprocessor IEEE TRANS ON COMPUTERS, 1977, Vol C-26, No 2, pp 138-146.

25. Wang, K. Stream oriented computation in recursive data flow schemes. MIT Project MAC, 1975, TM-68.

26. Davis, A. L. A data flow evaluation system based on the concept of recursive locality. AFIPS, 1979, pp1079-1086.

27. U.S. Patent No 3978452. System and method for concurrent and pipeline processing employing a data driven network. R. S. Barton, A. L. Davis, et al. Published 31 Aug 76.

28. Dennis, J. B. Data flow schemas. LECTURE NOTES IN COMPUTER SCIENCE. Springer-Verlag, 1972, Vol 5, pp 187-216.

29. Rodriguez, J. D. A graph model for parallel computations. MIT Project MAC, 1969, TR-64.

30. Keller, R. M., Lindstrom, G., and Patti, S. A loosely-coupled applicative multiprocessing system. AFIPS, 1979, pp 613-622.

31. Lecouffe, M-P. MAUD: a dynamic single assignment system. COMPUTERS AND DIGITAL TECHNIQUES, 1979, Vol 2, No 2, pp 75-79.

[119-2174]

2174
CSO: 1863

UDC 681.3

TASK ORIENTATION OF COMPUTER SYSTEMS BY MEANS OF ASSOCIATIVE LEARNING

Moscow AVTOMATIKA I TELEMEKHANIKA in Russian No 2, Feb 81 pp 156-163
manuscript received 23 Jul 80

IGNATUSHCHENKO, V. V., PRANGISHVILI, I. V., TRAKHTENGERTS, E. A. and FATKIN, Yu. M.

[Abstract] New methods are proposed in the article for improving the problem solving efficiency of computer systems through the utilization of technical, hardware and software systems and through learning, which leads to the replacement of some computations, necessary for solving problems by traditional methods, with more effective procedures by means of associative selection of the most efficient algorithms and problem-solving paramerters, and of special resource dispatching and assoicative data processing routines. The computers learn one step at a time how to do difficult practical tasks in real time. The learning procedure does not experience any changes during the solution of different kinds of problems, and the computers are oriented not at classes of problems, but toward a unified procedure for solving many classes of multidimensional problems by learning and utilizing experience. The learning procedure is based on the disclosure and identification of important attributes of a large number of solutions. The learning procedure and the hardware that efficiently implements it are examined. Future development of the system presumes the extensive application of associative data retrieval and processing methods for solving optimization problems. An assoicative parallel processor will be effective for searching for the most resemblant attribute vectors, which will substantially shorten data retrieval time. Figures 6; references: 10 Russian.
[127-7872]

ABSTRACTS FROM THE JOURNAL 'MICROELECTRONICS'

Moscow MIKROELEKTRONIKA in Russian Vol 10, No 1, Jan-Feb 81 pp 95-96

UDC 621.382

ARTIFICIAL INTELLIGENCE, ADAPTATION AND MICROELECTRONICS

[Abstract of article by Rastrigin, L. A.]

[Text] The subject of artificial intelligence is examined as a problem of discovering principles, and ways are described of solving it by adaptation methods in general and trial and error techniques in particular. It is shown that with a combination of adaptation and multidimensional extrapolation procedures it is possible to solve effectively the problem of synthesizing principles. The problem of the microelectronic implementation of these techniques is formulated in conclusion, providing effective solutions for poorly defined and weakly structured problems of artificial intelligence. Six illustrations, 15 references.

UDC 62-506.2.001.57

SYNTHESIS OF LARGE RECOGNITION SYSTEMS

[Abstract of article by Glaz, A. B.]

[Text] Examined are procedures for synthesizing large recognition systems, and illustrated by way of the example of the solution of model and practical pattern recognition problems is their effectiveness in the cases both of adequate, and of inadequate learning samples. The prospect of the design of such systems with microelectronics is emphasized. Three illustrations, 2 tables, 11 references.

UDC 621.382.8:001.57

MODERN CONTROL SYSTEMS AND MICROELECTRONICS PROBLEMS

[Abstract of article by Bukatova, I. L. and Yelinson, M. I.]

[Text] Analyzed in the article are the difficulties of solving problems of prediction, identification and control with the aid of computers within the framework of the traditional approaches to mathematical modeling. It is noted that methods that utilize abstract mathematical models (polynomials, differential,

integro-differential and other equations or equation systems) have two inherent basic deficiencies -- a fixed and inflexible form, and a functional dependence with limited dimension. Consequently the complexity (shape and size) of a mathematical model may turn out to be insufficient in real problems. To run in computers learning algorithms, in which mathematical models of optimum complexity are synthesized, as a rule requires considerable computer time expenditures and does not always satisfy the time frame of specific problems. These deficiencies of mathematical modeling can be overcome by means of physical (systems) implementation of learning algorithms, i.e., by means of the development of learning systems with a variable structure. Hardware models are synthesized in systems of this kind. Problems of microelectronics in the development of base elements with the necessary properties and satisfactory parameters are formulated in this connection. Twenty references.

UDC 681.3:007:62

PACKAGE OF DECISION MAKING ALGORITHMS AND CAPABILITIES OF MICROELECTRONICS

[Abstract of article by Erenshteyn, R. Kh.]

[Text] Aspects are examined of the synthesis of special recognition systems which utilize collective decision making techniques and parallel processing of incoming information. Problems are analyzed in which the development of such systems may significantly improve the effectiveness of their solution. A method is suggested for combining individual solutions into a collective one, utilizing the "experience" of the associative rules of the package and a specific recognizable situation. A method of synthesizing a specialized recognition unit is examined, based on the potential functions procedure. It is concluded that there is a need to develop a new element base for the described systems. Two illustrations, 11 references.

UDC 621.382

DEVELOPMENT OF EFFECTIVE RECOGNITION SYSTEMS AND MICROELECTRONICS PROBLEMS

[Abstract of article by Yelinson, M. I. and Sharov, A. M.]

[Text] Problems of the modeling of recognition processes are examined in the article. The existing capabilities are analyzed and compared and the prospects of the modeling of recognition systems with the aid of computers and with the aid of specialized hard models are also discussed. The advantages of on-line runs of recognition systems that offer an opportunity to parallelize both data processing and data storage and systems learning processes are explained. The need is demonstrated for synthesizing hardware models, in which both the memory systems and the learning control system are distributed. Examples are given of recognition problems, the computerized solution of which is still ineffective. Five illustrations, 16 references.

UDC 539.2

ANALYSIS OF SELF-ADAPTION IN ACTIVE DIFFUSIVE MEDIUM

[Abstract of article by Balkarey, Yu. I., Yevtikhov, M. G. and Yelinson, M. G.]

[Text] The formation and stability of special nonlinear laminations that occur in diffusive active (self-wave) diffusive media, are analyzed in a computer experiment. Chief attention is focused on the development of lamination in time from a local disturbance of a medium, initially in the homogeneous self-oscillation mode. The development process exhibits self-adaption of structural elements, which leads to a stationary state in a finite time. One-dimensional and two-dimensional cases are analyzed. The feasibility of using a medium of the described type as a special kind of memory is discussed. Seven illustrations, 14 references.

UDC 621.397.31

MICROELECTRONIC ANALOG PROCESSOR FOR MATRIX READING OF IMAGES

[Abstract of article by Svechnikov, S. V. and Popov, M. A.]

[Text] Examined is a processor for parallel reading and image input into a computer with parallel-organized computations. Reading is done with a matrix photoreceiver, controlled by an analog logic structure of neurons. An electrical diagram of a neuron and a detailed diagram of the processor are presented. Features of the application of the processor are discussed. Six illustrations, one table, 9 references.

7872
CSO: 1863

HARDWARE

SYSTEMS BASED ON EL'BRUS FAMILY OF COMPUTING MACHINES

Moscow PARALLEL'NYYE VYCHISLITEL'NYYE SISTEMY in Russian 1980 (signed to press 16 Sep 80) pp 201-214

[Section 4.2 "El'brus Family Systems" of book "Parallel Computer Systems" by Boris Arkad'yevich Golovkin, Izdatel'stvo "Nauka", 10,000 copies, 520 pages]

[Text] 4.2.1. Structural Conception of the El'brus Family Systems [56-60, 563] In development of high-performance systems of the El'brus family special attention should be given to solving the following problems:

1. insuring high reliability in problem solving;
2. overcoming the difficulties of writing and using working programs;
3. insuring system compatibility;
4. increasing the volume of main memory and the working speed of external memory;
5. interlinking territorially dispersed computer means using communications lines (constructing computer networks).

These are the key questions because unsatisfactory solutions to these problems have held back the development and introduction of large computer systems. The following discussion covers ways to solve these problems.

Reliability. The requirements of high problem-solving reliability involve insuring guaranteed receipt of reliable results within a certain given time interval and insuring guaranteed storage of large volumes of information for extended periods of time.

It is not possible to solve the reliability problem entirely simply by increasing the reliability of machine elements because higher and higher requirements are being imposed for expanding the processing capabilities of processes, increasing the capacity of main memory, and improving the

supply of peripheral equipment for machines. When these requirements are met the reliability safety margin of new elements is usually spent and the reliability of the new machines often remains a problem area. A fundamental solution to the reliability problem can only be obtained by structural means. One of these techniques is to use the modular principle of constructing computer systems with full hardware monitoring in each module.

A computer system consisting of identical modules of each type (processors, main and external memory, and input-output units) can in principle provide any required probability of receiving reliable results from problem-solving in a given time interval. The level of this probability is determined, where other conditions are equal, by the depth of redundancy. In this case the reliability of data storage is insured by redundancy at different levels of the memory hierarchy, with due regard for operational use of the data and the volume of data being stored.

The multiprocessor modular principle of organizing computer equipment has other advantages as well. For example, it makes it possible to construct computer systems that overlap a broad productivity range and have the required ratio between productivity and the capacity of main and external memory.

Writing and using working programs. If appropriate steps are not taken during the development of the structures of computer systems, it will probably be impossible to program many timely problems because of their complexity. In view of the importance of high-level languages and the internal structure of control and command systems of the computer system for easing the work of programming complex problems, it is wise to build the systems to provide the highest efficiency in performance of programs written in the high-level language. Two basic requirements can be made of such a machine language. It should not include any information except the algorithm description proper, and it should provide maximally compact notation of the algorithm as a program in this language. In other words, the program in machine language should be a concrete realization of the algorithm in the most compact form possible without information on control of the physical resources of the computer system.

To meet the first requirement the machine language should not contain information on the type of memory, the physical addresses of operands, and information that directly or indirectly indicates the units in which processing must be done. The problem of excluding the physical addresses of external memory in machine languages has in fact been solved. In view of the comparatively slow speed of external memory, the corresponding functions are performed by the programs of the operations system. As for excluding the addresses of main and register memory, possible solutions could be organizing mathematical (software) and corresponding physical (hardware) main memory in the form of segments of any necessary length as system units of information and dynamic distribution of register memory accordingly by hardware means. The use of basing and software memory brought the solution to the problem closer, but did not solve it. In fact, basing provides a direct correspondence between the limited linear domain of addresses used and the equivalent domain of

physical memory. When addressing is done using mathematical memory, the memory itself serves as a resource and, moreover, the requirement of compactness is not met because of the redundancy of information on mathematical addresses of the program module.

In order to preclude information on the type of processing unit to use from the program in machine language, it is necessary to use, in addition to the size code, an indicator of its type like the descriptions of types in algorithmic languages. Using the types of quantities the machine dynamically selects the appropriate processing units.

To meet the second requirement (compact notation of algorithms in machine language) the coding of data on addresses and statements must be optimized. For this purpose it is advisable to realize directly in the hardware the established methods of algorithm description and organization of computing processes used in the high-level languages and operations systems, first of all to realize the statements and procedures. In this case the program in machine language becomes a hierarchically structured program, the machine language (as the primary variation) becomes a language with unaddressed indication of operands by means of names, while the corresponding units of the machine perform the procedures of particular operation (reading, writing, arithmetic operations, and others) by hardware means. Broadening the capacities of the hardware does not reduce speed because the time required to perform purely repetitive subsidiary operations is reduced. In this case, of course, the equipment of the computer system as a whole performs the more complex logical functions.

Compatibility. Hardware realization of high-level languages helps significantly to solve the problem of compatibility of machines of different generations because programs in high-level languages can be retranslated to the languages of newly developed machines. In addition, even when a high-level machine language has been fixed, it is possible to build new machines with a higher degree of freedom to alter structural decisions than would be the case if the traditional language were preserved or expanded. To insure program compatibility, however, it is necessary to preserve the functions of the old operations systems related to interaction with working programs and to keep the structure of the old archives. Solving this problem requires the introduction and standardization of basic procedures for organizing the structure and functioning of operations systems and archives, which will in turn make it possible to move on to realizing them in the structure of the machines.

Memory problems. The structure and basic characteristics of large high-performance computer systems are determined in large part by their memory specifications. The principal problems here are increasing the volume and speed of main memory and increasing the average speed of exchange between the central part of the computing system and large-volume external memory. Solving the first problem requires both qualitative and quantitative improvement of the basic elements. One of the main ways to solve the second problem, in addition to improving the characteristics of external memory, is to introduce intermediate memory, for example introducing intermediate memory on disks with a heading on the track or the ferrites.

This is often done and improves the coordination of working speeds between main and external memory.

In addition to improving memory characteristics, an effort should also be made to reduce the load on memory, above all main memory, by structural decisions. Realization of a high-level machine language reduces the length of the program and the frequency of references to memory, that is, makes it possible to conserve main memory to store programs and, at the same time, use its speed resources more economically. In some segments of algorithms the length of programs is reduced by one-half or two-thirds and the number of references to main memory is cut dozens of times. Hardware realization of the most regular functions of the operations systems also reduces the load on main memory.

The use of high-speed memory in addition to reducing access time to data for processors greatly decreases the frequency of references to main memory, thus improving the coordination of the working speeds of processors and main memory. This is particularly important in multiprocessor systems because in them several processors are working with a common main memory. It should be observed that the construction of a single high-speed memory for all processors is unacceptable because in this case the dynamic resource of the high-speed memory will be a bottleneck in the entire system and, moreover, access time to this memory will prove too great because of the relatively long communications lines between processors and the single high-speed memory. At the same time, the accepted structure of individual high-speed memory devices for each processor makes it more difficult for them to work with common data.

It should also be observed that a main memory unit with large capacity and high speed cannot be realized in the form of a single block because of technical limitations. Therefore, the main memory is distributed over a number of blocks. Although this is inevitable from this point of view, it also corresponds to the general modular principle.

Building computer networks. To interconnect territorially dispersed computing equipment using data transmission lines, it is advisable for the computer systems to include special processors for data transmission and receiving and give them the primary functions of interacting with the various other installations by communications lines. The processor for data transmission and receiving should operate with its own memory and be capable of effectively maintaining the work of the computer network. Then the basic computing systems of the centers in the network will be able to adapt to different conditions of organization and development of the computer network without having to change their basic programs and equipment.

Thus, the basic conceptions of the organization of computing systems for the El'brus family are: the modular principle of constructing systems with many identical basic processors and shared main memory; a system of processor commands in a high-level language; hardware realization of regular, established methods of work by operations systems and principles of archive construction; multilevel memory with expanded main memory and a relatively fast large-capacity intermediate memory; specialized processors

for data transmission and receiving which are introduced to work with various installations through telephone and telegraph channels. Figure 4.15 below shows a simplified structural diagram of this kind [56].

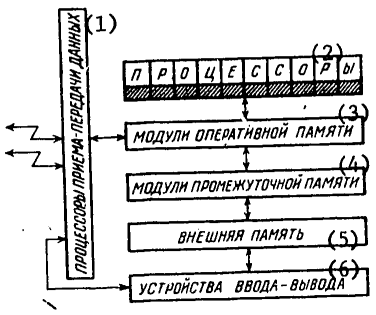


Figure 4.15. Simplified Structural Diagram of El'brus Systems.

Key: (1) Processors for Data Transmission and Receiving;
(2) Processors;
(3) Main Memory Modules;
(4) Intermediate Memory Modules;
(5) External Memory;
(6) Input-Output Units.

4.2.2. Structure of the El'brus Family Systems. Let us consider the structure of El'brus multiprocessor computing complexes, directing attention to efficient use of equipment, insuring maximum productivity, the reliability of redundancy structures, the possibility of adding to the configuration, and the possibility of adaptation to the problems being solved.

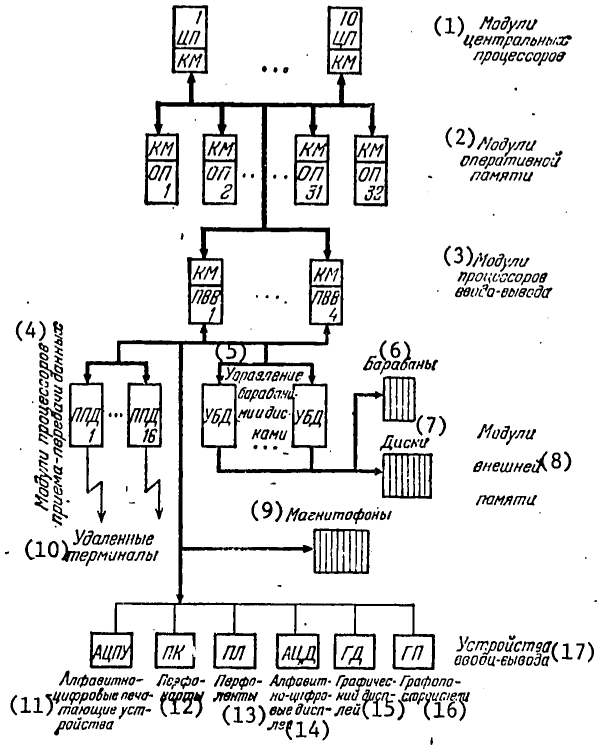
Figure 4.16 shows the general structural diagram of a multiprocessor computing complex (MCC). Some of the elements of the MCC have been omitted in this diagram and the connections are shown in generalized form.

Appropriate explanations are given at later points in the text (see Figure 4.16, next page).

Let us consider the structure of the El'brus MCC [59, 348]. The El'brus MCC, constructed on the modular principle, contains modules of central processors, main memory, input-output processors, processors for data transmission and receiving, external memory on magnetic drums, disks, and tape, and input-output units.

The modules of the central part of the MCC, in particular the modules of the central processors, main memory, and input-output processors, are interlinked by means of a central commutator whose modules are distributed at the appropriate functional modules. The realized module of the central commutator has a dimensionality of 4 × 14 and joins four modules of main memory with all the modules of the central processors and input-output processors. The processors for transmitting and receiving data, the intermediate and external memory units, and the input-output units are connected to the central part of the system through the input-output processors.

Figure 4.16. Structural Diagram of the El'brus Multiprocessor Computing Complex.



Key: (1) Modules of Central Processors;
(2) Main Memory Modules;
(3) Modules of Input-Output Processors;
(4) Modules of Processors for Data Transmission and Receiving;
(5) Control of Drums and Disks;
(6) Drums;
(7) Disks;
(8) External Memory Modules;
(9) Tape Recorders;
(10) Remote Terminals;
(11) Alphanumeric Printers;
(12) Punched Cards;
(13) Punched Tapes;
(14) Alphanumeric Display;
(15) Graphic Display;
(16) Graph Plotters;
(17) Input-Output Units.

External memory is connected to the input-output processor channels through units that control the magnetic drums, disks, and tape accordingly. The telephone and telegraph communications lines are connected to the processors for data transmission and receiving, which do not have their own system of commands or internal memory, through adapters and group interlinking devices. Thus, information from a remote terminal, for example, enters the central part of the system, in order, through the communications line, adapter, group interlinking device, and processor for data transmission and receiving.

The modules of the system work in parallel and are independent of one another; system resources are dynamically distributed by the operations system. Modules of the same type are identical and independent (with respect to both commutation and power supply). All the modules, including the central commutator assemblies related to them, have complete hardware monitoring whose circuits produce a malfunction signal for the appropriate module at the appearance of even one isolated error in the computation process. The operations system, having received such a signal, switches the malfunctioning module out of the work mode and uses the hardware reconfiguration system to change it to the repair configuration. After the module is restored it can be switched back to the working configuration by the operations system.

Thus, redundancy is accomplished at the level of the functional modules and devices of each particular type. When any module goes down its work can be switched to other modules of the same type, which lowers the working characteristics but avoids a shutdown of the entire MCC. The MCC can be adjusted to solve different classes of problems by selecting an appropriate configuration. In other words, the multiprocessor modular structure provides flexibility and the assigned levels of reliability and survivability for the MCC.

The organization of computations is based on a stack (memory domain with reading of the last written word). The stack can be used both for computing expressions and for dynamic distribution of memory by procedure names. Each procedure in the computation process has its own domain of directly-addressed data. The address of the beginning of this domain is the base of the procedure. It also has an address context in the form of a domain of name prominence, which comprises the names proper and the names of the procedures covered in conformity with the block structure of the program.

Each word has a tag which indicates the type of data (whole, substantive, set, descriptor, address, procedure symbol, and the like) and the data format (32, 64, or 128 bit positions).

The system of commands is based on the no-address principle, which makes it possible to exclude the addresses of fast registers from the command code. Addressing is done by an address pair in the form of a code on the lexicographic level of the given procedure (level of inclusion) which determines the base of this procedure and the ordinal number of the word in the data stack of the same procedure, which determines its displacement relative to the base (indexing). The most frequently used commands are one, two, and three bytes long; commands with multiple bytes also occur.

The data being processed is represented in arrays described by descriptors. The codes of the commands are loaded in physical memory while the data proper is "housed" in mathematical memory.

The computing process may be active or passive. In the first case when the processor is working the computing process is represented in hardware terms by the stack, which is usually located in main memory, the stack indicator, the base registers, high-speed memory, and other central processor equipment. In the second case, when the process is waiting for some event, it is reflected in hardware terms only by the data in memory. A separate stack may be assigned for each procedure to perform parallel computations. Several processors may be active at one time; in this case any processor can work on any stack. Semaphores are used to synchronize the processes; the system of commands contains operations for "open semaphore" and "close semaphore."

Suppose that all the processors are busy and there are also processes ready for performance in the queue of processes waiting for a processor to become free. Then, as soon as a certain active process turns passive for some reason or is completed, the corresponding processor becomes free, goes to the queue of ready processes, and begins to perform the first one. Dispatcher control of the work of the MCC is accomplished by the operations system.

The high-speed memory is distributed to the central processors and broken down by functional characteristics in each processor into fast virtual registers for buffering computation of statements (top of the stack), directly-addressed memory for buffering local data of the process (continuous linear storage segment of the stack), associative memory to buffer the universal data of the process, associative memory to buffer commands, and an associative memory to buffer arrays. The command buffer speeds up their selection, especially if commands are repeated often (cycles), and the buffer of arrays insures that data is fed on time.

The presence of a number of processes and the possibility of increasing the number (among the other advantages of modular structure) insures high system productivity. The tight spot is the shared main memory whose efficient use is determined largely by the hardware and software means of distributing it.

The El'brus MCC adopted hardware recursion, which makes it possible to have one copy of the programs in main memory, and thus saves storage space.

When programs are translated, homogeneous information of each type is combined and the segments obtained are described by their own descriptors with discreteness of up to a word. The volume of physical memory allocated is determined entirely by the requirements of the descriptors. The mathematical memory is broken into pages of 512 words, and when it is "filled" it usually contains incomplete pages (a segment whose length is not a multiple of 512 does not completely fill a page). The mathematical memory is used just once, which makes it possible without conflict to copy data into external memory, restore it, and redistribute it. Segmental distribution of physical memory without rigid boundaries is oriented to exchange with units of the system at a higher level of the hierarchy, while page

distribution of mathematical memory is oriented to exchange with lower-level units.

Organizing computations on the basis of the stack makes it possible to call up data at practically the last moment and to free memory at any moment after completion of the corresponding computations.

The steps we have listed raise the efficiency of use of main memory, but they make the operations system and processor access to data more complex. This can be compensated for, in turn, by introducing a high-speed memory.

The input-output processor performs the basic exchange functions in the El'brus MCC, including the organization of exchange both between main memory and external units and between pairs of non-external units. It has a buffer memory and the essential arithmetic and logical circuits and works within the work setting of the central processor on assignment from the processor, which takes the form of a reference to an external device "written" by the operations system. The list of these requests is stored in main memory. When performing the next request, the processor sets the appropriate external unit on exchange, transmits the data, completes the exchange, and corrects the list of requests for exchange. If there are several routes for data exchange (the external unit can work through any of four processors and any corresponding units for control of external units), the choice of the path of exchange is made as the corresponding equipment becomes free. All the functions of controlling external equipment, including dispatcher control, reduncancy, and selecting the correct channel, are done by hardware. This makes it possible to save main memory space and significantly free central processes from interrupts by external units.

In the processors for data transmission and receiving, unlike the input-output processors, the program method of interaction with other units has been adopted because the number of these outside units is large and there are many different types. The software of the first processors is organized on the modular principle so that all that is required to connect a new installation is to add an essentially new program module with a description of this unit.

The software of the MCC is based on the principle of achieving highly efficient counting by means of the flexibility and adaptability of the computer system. This principle is realized by integrated development of system hardware and software [24]. In this case a comparatively small set of basic software capabilities is carefully chosen and the optimally effective means of combination are worked out so that the programmer can design the particular programs efficiently. In this case the computer system with its software adapts, so to speak, to the problems being solved. This is characteristic of the El'brus MCC, not the opposite situation where the algorithm must be modified within the rigid framework of an inflexible computing system.

The El'brus MCC is well adapted for using this principle because it has a high-level machine language, is flexible, realizes numerous dynamic mechanisms, and has a high degree of virtuality. For example, the mechanisms of dynamic distribution of resources are realized in hardware and in the

operations system, that is, on the internal levels of the system, while the distribution of processors is accomplished automatically on a dynamic basis.

This results in a simplification of programming, economy of system resources and greater problem-solving efficiency, precluding the specific characteristics of controlling assignments, and simplification of both system start-up and re-start-up and the system operator's work.

The software of the El'brus MCC, like its hardware, is constructed on the modular principle. Both the programs themselves and the data are structured. In addition, steps were taken during the preparation of software to make system programming and programming by users closer. In both cases programming is done in a high-level language, and practically all user mechanisms are realized by software. This makes it possible to provide the user with well-tested and efficient programming mechanisms as early as possible and improve the technology of both the system and user programming. It also makes it possible to reduce the volume of software [24].

The first model of the El'brus family is the El'brus-1 MCC. This MCC has the following basic specifications [278].

The El'brus-1 has a productivity of 1.5-12 million operations a second and a main memory capacity between 567,000 and 4,608,000 bytes depending on its set of equipment. The set may include 1-10 central processor modules, 4-32 main memory modules, 1-4 input-output processor modules, and 1-16 modules of processors for data transmission and receiving (four for each input-output processor), as well as external memory modules on magnetic drums, disks, and tape and various input-output units.

The central processor operates with numbers in 32, 64, and 128 bit format and with alphanumeric information. It provides practically unlimited virtual memory (2^{32} words). Addition with a fixed decimal point is performed in 520 nanoseconds, while with a floating decimal point it takes 780 nanoseconds. Multiplication of 32-position numbers takes 780 nanoseconds, while for 64-bit numbers it is 1,300 nanoseconds. Logical operations and operations with fields are accomplished in 520 nanoseconds. The maximum productivity of the central processor is 1.5 million operations a second. The logical part of the processor uses integrated circuits.

The El'brus-1 MCC can perform BESM-6 user programs applicable to the Dispak system. To do this one of the central processors is replaced with a specially developed processor that performs the BESM-6 system of commands. This special processor may have a productivity of 3 million operations a second. It performs addition in 240 nanoseconds, multiplication in 400 nanoseconds, and logical operations in 80 nanoseconds.

The main memory is built on cores and is connected to the central processors and input-output processors through the main memory commutator. Each commutator connects four main memory modules with overlapping access cycles to not more than 14 central and input-output processors (10 of the former and four of the latter). Four memory modules have a total capacity of 64,000 72-bit words. The modules consist of memory blocks each of which

has a capacity of 4,000 36-bit words. Cycle time is 1.2 microseconds and access time is 0.5 microseconds.

The input-output processors have hardware that performs the basic algorithms of input-output dispatcher control of the operations system and controls the work of peripheral equipment. Each such processor receives outside interrupts and interrupts from the 10 central processors; it may itself interrupt the work of these processors. It is connected with main memory by eight channels as well as with external memory, the input-output unit, and the processors for data transmission and receiving.

The input-output processor includes the following blocks: (1) a fast channel block which connects through the four channels of the block with up to 64 accumulators on magnetic drums and replaceable magnetic disks and permits four of them to work at the same time; (2) a standard channels block which connects it through the 16 channels of the block to up to 256 memory units on magnetic tape and input-output devices and permits 16 of them to work simultaneously; (3) an interlinking block and optimizer which connects it through one channel of the block to four processors for data transmission and receiving operating simultaneously and, through the optimizer of the block, to 16 magnetic drum memory units which reduces average access time to them 4-6 times. The number of blocks of fast and standard channels in the input-output processor is determined by the makeup of the equipment; each processor has one interlinking block and one optimizer.

The maximum speed of exchange between one input-output processor and main memory is 36 million bytes a second; the exchange speeds through a fast channel, a standard channel, and a channel for the processor for data transmission and receiving are 4, 1.3, and 1 million bytes a second respectively.

The system ordinarily uses standard YeS [Unified System] devices for external memory and input-output units.

The processor for data transmission and receiving with its own main memory, an elaborate system of commands oriented to work with remote users, and a flexible operations system has a productivity of 700,000 operations a second and, using group interlinking devices, can serve up to 160 telephone or telegraph lines. The group interlinking unit can serve up to 16 telegraph or telephone lines and insures program adaptation to different types of data transmission systems and dynamic monitoring of the line. The remote control subsystem as a whole is designed for 2,560 channels.

Four standard sets of equipment for the El'brus-1 MCC are recommended. They contain 1, 2, 4, and 10 central processors, main memory capacities of 576,000, 1,152,000, 2,304,000, and 4,608,000 bytes, and other equipment. Their productivity levels are 1.5, 3.0, 5.5, and 12.0 million operations a second.

The software of the El'brus-1 MCC includes a programming system, a single operations system, a system of standard and service programs, a remote control system, and test and diagnostic programs.

The programming system gives users the following high-level languages: Algol 60, Fortran, Cobol, PL/1, Algol 68, Simula 67, Pascal, and El'brus Autocode. It debugs programs at the input language level, diagnoses program errors in compiling and execution, combines and segments programs, and provides multiple entries to programs and various capabilities for security, editing, and the like.

The operations system provides multiprogram modes of close and distant batch processing, a time-sharing mode, and terminal processing. It controls the work of the processors, synchronizes them, performs dynamic distribution of system resources, provides virtual memory, and distributes physical memory into segments without rigid boundaries to the precision level of one word. The operations system also provides automatic reconfiguration of the MCC, reconstruction of files, and restarting of problems and the system in order to give it functional reliability and various other capabilities. The small volume of the resident part of the operations system in machine language should be noted [278]. Despite the broad assortment of functions in the operations system, it has a fairly small volume. The total volume of the operations system and translater from autocode is 250,000 lines [25].

An experimental batch of El'brus-1 MCC's had been built by the beginning of 1978. Further development of its principles of structure made it possible to work out and move on to introducing the next model, the El'brus-2, into production. It has a productivity of more than 100 million operations a second. Work has begun to design computer systems with even higher productivity. The El'brus-2 MCC uses large integrated circuits; its modules are functionally identical to El'brus-1 modules. The software of the El'brus-1 MCC can be used in the El'brus-2 MCC without changes [61].

The systems of the El'brus family are classified as MKMDS/VsOrPk's — systems with multiple flows of commands and data, processing by word, and a homogeneous structure with a high degree of interlinking and (with certain special characteristics) crossing links in the central part.

In conclusion we should observe that the areas of work today in building systems using the SM, YeS, and El'brus families cover, with overlapping boundaries, practically the entire spectrum of small-medium, medium-high, and high-superhigh productivity values.

4.23. Bibliographic Note. The design concept, structure, and basic characteristics of the El'brus systems as well as prospects for the development of these systems are described in [22, 56-61, 278, 304, 347, 348, and 563]. Work [218] has a short description of El'brus systems, and a very short description is also found in [447]. The principles of construction, realization, and basic characteristics of the software of these systems are presented in [23-25, 29, 144, 145, 181, 199, 200, 320-324, 375, 376, 394, and 421-423]. Some basic data on the El'brus MCC and corresponding references are listed in [110].

FOOTNOTES

22. Babayan, B. A., "Struktura Sovremennykh EVM" [The Structure of Contemporary Computers], Moscow, ITM i VT AN SSSR, 1975, p 14.

23. Babayan, B. A., "Software of Multiprocessor Systems," in "Sbornik Tezisov Dokladov Vsevoyuznoy Nauchno-Tekhnicheskoy Konferentsii po Problemam Elektronnoy Vychislitel'noy Tekhniki EVM-76" [Collection of Abstracts of Reports from the Computer-76 All-Union Scientific-Technical Conference on Problems of Electronic Computing Equipment], Part 1, Moscow, NIITsEVT, 1976, p 72.

24. Babayan, B. A., "Osnovnyye Printsipy Programmnogo Obespecheniya MVK El'brus" [Basic Principles of Software for the El'brus Multiprocessor Computer Complex], Moscow, ITM i VT AN SSSR, 1977, preprint No 5, 11 pages.

25. Babayan, B. A., "The Technology of Programming the Software of the El'brus-1 Multiprocessor Computer Complex," in "Tekhnologiya Programmirovaniya. Tezisy Dokladov I Vsesoyuznoy Konferentsii. Plenarnyye Doklady i Obshchiye Materialy" [Programming Technology. Abstracts of Reports from the 1st All-Union Conference. Plenary Reports and General Materials], Kiev, IK AN USSR, 1979, pp 36-37.

29. Bezhanova, M. M., "Kompilyator s Yazyka IL/1 dlya MVK El'brus-1" [Compiler from IL/1 Language for the El'brus-1 Multiprocessor Computer Complex], ITM i VY AN SSSR, Novosibirsk Branch, 1978, Preprint No 24, 16 pages.

56. Burtsev, V. S., "Perspektivy Razvitiya Vychislitel'noy Tekhniki: Doklad na Yubileynoy Nauchno-Tekhnicheskoy Konferentsii, Posvyashchennoy 25-Letiyu Instituta Tochnoy Mekhaniki i Vychislitel'noy Tekhniki AN SSSR" [Prospects for the Development of Computer Technology: Report at the Anniversary Scientific-Technical Conference Dedicated to the 25th Anniversary of the Institute of Precision Mechanics and Computer Technology of the USSR Academy of Sciences], Moscow, ITM i VT AN SSSR, 1975, 29 pages.

59. Burtsev, V. S., "Printsipy Postroyeniya Mnogoprotsessornykh Vychislitel'nykh Kompleksov El'brus" [Principles of Construction of El'brus Multiprocessor Complexes], Moscow, ITM i VT AN SSSR, 1977, Preprint No 1, 53 pages.

60. Burtsev, V. S., "Tendentsii Razvitiya Vysokoproizvoditel'nykh Sistem i Mnogoprotsessornyye Vychislitel'nyye Kompleksy" [Trends in the Development of High-Performance Systems and Multiprocessor Computer Complexes], Moscow, ITM i VT AN SSSR, 1977, 28 pages.

61. Burtsev, V. S., "The Computer: The Series of Generations" PRAVDA 4 April 1978, No 94(21794).

110. Golovkin, B. A., "Methods and Means of Parallel Data Processing," in "Itogi Nauki i Tekhniki. Teoriya Veroyatnostey. Matematicheskaya Statistika. Teoreticheskaya Kibernetika" [Scientific and Technical

Results. Probability Theory. Mathematical Statistics. Theoretical Cybernetics], Vol 17, Moscow, VINITI, 1979, pp 85-193.

144. Dyatlova, L. G., Kozhukhina, S. K., Lesovaya, I. F., Pen'kova, L. N., Petrashkova, N. Yu., Sidorova, N. A., Khmel'nitskaya, S., and Shishkova, N. A., "Compilers from Algol-60 and Fortran for High-Performance Computers," in "Sbornik Tezisov... op. cit. (fn. 23), pp 129-131.

145. Dyatlova, L. G., Kozhukhina, S. K., Lesovaya, I. F., Pen'kova, L. N., Petrashkova, N. Yu., Romanishina, N. M., Sidorova, N. A., and Shishkova, N. A., "Realization of Algol-60 and Fortran in the El'brus-1 Multiprocessor Computer Complex," Moscow, ITM i VT AN SSSR, Novosibirsk Branch, 1978, Preprint No 21, 16 pages.

181. Ivanov, A. P., and Semenikhin, S. V., "Operatsionnaya Sistem MVK El'brus-1" [The Operations System of the El'brus-1 Multiprocessor Computer Complex], Moscow, ITM i VT AN SSSR, 1977, Preprint No 2, 31 pages.

199. Katkov, V. L., "Programmnoye Obespecheniye MVK El'brus-1 i El'brus-2 i NF ITM i VT" [Software of the El'brus-1 and El'brus-2 Multiprocessor Computer Complexes at the Novosibirsk Branch of the Institute of Precision Mechanics and Computer Technology], Moscow, ITM i VT AN SSSR, Novosibirsk Branch, 1978, Preprint No 25, 15 pages.

200. Katkov, V. L., "Development of Machine Graphics Software for the El'brus Multiprocessor Computer Complex" AVTOMETRIYA 1978, No 5, pp 12-18.

218. Korolev, L. N., "Struktury EVM i Ikh Matematicheskoye Obespecheniye" [The Structure of Computers and Their Software], Moscow, Nauka, Glavnaya Redaktsiya Fiziko-Matematicheskoy Literatury, 1974, 256 pages; 2nd edition, 1978, 351 pages.

278. "Mnogoprotsessornyy Vychislitel'nyy Kompleks El'brus-1: Prospekt" [The El'brus-1 Multiprocessor Computer Complex: Prospectus], Moscow, ITM i VT AN SSSR, 1977, 19 pages.

304. "The New Soviet Computer Complex," MOSKOVSKAYA PRAVDA 8 April 1978, No 82 (17704).

320. Pentkovskiy, V. M., "Base Programming Language for Multiprocessor Systems," in "Sbornik Tezisov..., op. cit., (fn. 23), p 73.

321. Pentkovskiy, V. M., "Osnovnyye Kharakterisitiki Avtokoda MVI El'brus" [Basics Characteristics of the Autocode of the El'brus Multiprocessor Computer Complex], Moscow, ITM i VT AN SSSR, 1977, Preprint No 6, 29 pages.

322. Pervin, Yu. A., and Khoperskov, A. Ye., "Compiler from Cobol," in "Sbornik Tezisov..., op. cit., (fn. 23), p 132.

323. Pervin, Yu. A., and Khoperskov, A. Ye., "Kompilyator s Yazyka KOBOL dlya MVK El'brus-1" [Compiler from Cobol for the El'brus-1 Multiprocessor Computer Complex], Moscow, ITM i VT AN SSSR, Novosibirsk Branch, 1978, Preprint No 23, 20 pages.

324. Perekatov, V. I., Larin, Ye. M., and Osinov, A. V., "Sistema Teleobrabotki MVK El'brus-1" [The Remote Processing System of the El'brus-1 Multiprocessor Computer System], Moscow, ITM i VT AN SSSR, 1977, Preprint No 4, 26 pages.

347. Pshenichnikov, L. Ye., Solovskoy, A. A., Zakhvatov, M. V., and Kol'tsova, S. L., "The Organization of Input-Output in Multiprocessor Systems," in "Sbornik Tezisov..., op. cit. (fn. 23), p 31.

348. Pshenichnikov, L. Ye., and Sakhin, Yu. Kh., "Arkhitektura Mnogoprotsessornogo Vychislitel'nogo Kompleksa El'brus" [The Architecture of the El'brus Multiprocessor Computer Complex], Moscow, ITM i VT AN SSSR, 1977, Preprint No 7, 40 pages.

375. Semenikhin, S. V., and Zotov, S. A., "Control of Computing Processes in Multiprocessor Systems," in "Sbornik Tezisov..., op. cit. (fn. 23), p 31.

376. Semenikhin, S. V., and Molchadskiy, V. D., "Control of Assignments in Multiprocessor Systems," in "Sbornik Tezisov..., op. cit. (fn. 23), p 75.

394. Torchigin, V. N., "Sistema Faylov MVK El'grus-1" [The File System of the El'brus-1 Multiprocessor Computer System], Moscow, ITM i VT AN SSSR, 1977, Preprint No 3, 39 pages.

421. Tsang, F. R., "The Operations System of the VS Special Processor, Compatible with the Dispak Operations System," in "Sbornik Tezisov... op. cit. (fn. 23), p 126.

422. Tsang, F. R., "Operatsionnaya Sistema Spetsprotsessora MVK El'brus-1, Sovmestimaya s OS DISPAK" [The Operations System of the Special Processor of the El'brus-1 Multiprocessor Computer Complex, Compatible with the Dispak Operations System], Moscow, ITM i VT AN SSSR, Novosibirsk Branch, 1978, Preprint No 20,13 pages.

423. Tsang, F. R., and Chinin, G. D., "Printsipy Postroyeniya i Realizatsii Operatsionnoy Sistemy Spetsprotsessora MVK El'brus-1" [Principles of Organization and Realization of the Operations System of the Special Processor of the El'brus-1 Multiprocessor Computer Complex], Moscow, ITM i VT AN SSSR, Novosibirsk Branch, 1978, Preprint No 22, 12 pages.

447. Yakubaytis, E. A., "Arkhitektura Vychislitel'nykh Sistem i Setey" [The Architecture of Computer Systems and Networks], Riga, In-ta Elektroniki i Vychislite'noy Tekhniki AN Latv. SSR, 1978, IEVT POI, 37 pages.

563. Burtsev, V. S., "High-Performance Computers and Multiprocessor Computer Systems Development Trends," PROC. IFAC CONGR., Helsinki, 1978.

[146-11,176]

11,176
CSO: 1863

THE M-10 COMPUTER SYSTEM

Moscow PARALLEL'NYYE VYCHISLITEL'NYYE SISTEMY in Russian 1980 (signed to press 16 Sep 80) pp 307-311

[Section 6.2 of book "Parallel Computer Systems" by Boris Arkad'yevich Golovkin, Izdatel'stvo "Nauka", 10,000 copies, 520 pages]

[Text] 6.2. The M-10 Computer System [196, 197, 881]*

The development of the M-10 system was oriented to using the parallelism of a set of objects, which is typical for the problems of processing arrays of data that describe a large number of identical or almost identical objects, and the parallelism of related operations in algorithms, which consists in the fact that related operations may be performed simultaneously if the results of a certain operation are not used as input data in one or several subsequent operations.

The general conception of the M-10 system is as follows.

The processors (Пр) are grouped in a line of i processors in such a way that each line is linked to one corresponding group of control signals of the control unit and is controlled by a common op code. It is not mandatory here that the processors of the line be identical or that they receive identical control signals. For efficient use of a line the command system should have group operations on arrays, vectors of numbers, and vectors of functions. It is advisable to organize the main memory of the processors in the form of a common basic memory of the line with controlled broad format for access (from 1 to i words).

Several lines can be joined in a bundle. This may consist, for example, of j lines arranged in such a way that all the lines are connected with a common control device for the bundle (У), each command of which contains j op codes, one for each line of the bundle. The mode of work is

* Material from M. A. Kartsev's article "The M-10 Computer" (DOKLADY AN SSSR, 1979, Vol 245, No 2, pp 309-312) was used in writing section 6.2. The author did not become aware of this article until the manuscript of the book was nearly completed. Brief information on the M-10 system can also be found in the report "Five Million a Second" (SOTSIALISTICHESKAYA INDUSTRIYA 13 Sep 79, No 211-3103).

synchronous because the lines are controlled by a common bundle control unit, even though the lines have different op codes. It is not mandatory here that all j lines of the bundle be identical or have the same number of processors. To continue, k bundles may be joined into a combined-type synchronous system (see Figure 6.4 below) which has a common working rhythm that is assigned by means of a common control system. Such a system may work as asynchronous in certain segments of the parallel program, but when necessary the system is put in a synchronous working mode by program means with an indication of the operations for the processors in each cycle of fulfillment of the commands.

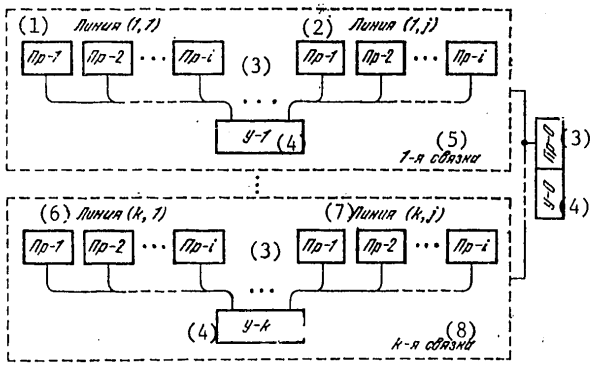


Figure 6.4. Structural Diagram of a Combined-Type System

Key: (1) Line (1, 1);
(2) Line (1, j);
(3) [Processors: "Пр-1" — Processor No 1... etc.];
(4) [Control Units: "У-1" — Control Unit No 1... etc.];
(5) First Bundle;
(6) Line (k, 1);
(7) Line (k, j);
(8) Bundle k.

Now let us look at the M-10 system itself.

The M-10 is a multiprocessor synchronous system with an average productivity of more than 5 million operations a second and internal memory capacity of 5 megabytes (1,310,720 32-bit words). It has a cycle of 1.8 microseconds. The system operates with numbers in three formats. The numbers of the first format occupy 16 bit positions (a half-word) and are entirely numbers with fixed decimal points (whole numbers and fractions). The numbers of the second and third formats occupy 32 bit positions (a word) and 64 bit positions (a double word) respectively and can be either numbers with a fixed decimal

decimal point (whole numbers or fractions) or numbers with a floating decimal point. In the latter case eight bit positions are allocated in the order for numbers of both formats with floating decimal points. Numbers with 128 bit positions may also be used; an incomplete list of operations of the M-10 system is used for operating with them.

The main functions of data processing in the system are performed by two lines, which can be rearranged by program means, of the processors. Depending on the op code each line is either eight 16-bit processors, or four 32-bit processors, or a pair of 64-bit processors which perform the same operation on (different) data. The processors of a line are combined in a single vector process by means of definite op codes. For example, when computing the scalar product of vectors the processors of a line in one cycle of the system perform by-pair multiplication of eight pairs of 16-bit numbers or four pairs of 32-bit numbers, sum the products of these multiplications, and add them to the sum accumulated in the preceding cycle. The two lines may perform the same operation or different operations at one time.

Computation of the results of operations on numbers is accompanied by the production of up to five lines of values of Boolean variables in the lines. Each value describes one of the operands or the result of an operation. For example, when performing addition of eight pairs of 16-bit numbers, four pairs of 32-bit numbers, or two pairs of 64-bit numbers, the lines produced have eight, four, or two values of Boolean variales respectively, containing signs to indicate: (1) overflow; (2) equality of the components; (3) one component exceeding the other; (4) a result equal to zero; (5) a negative result. These indicators may be transmitted directly or through memory to a special processor that processes lines of Boolean variables. It has a full set of logical operations on Boolean variables and functions simultaneously with the lines of the main processors.

The lines of values of Boolean variables received directly during the performance of operations on numbers in the lines of the primary processors, when performing operations in the special processor, or resulting from fetching from memory may be used for provisional transfers of control and to impose masks on the primary processors.

The internal memory of the system contains a main operational memory with a capacity of 512,000 bytes (131,072 words), a main permanent memory with the same capacity, and a large operational memory with a capacity of 4,000,000 (1,048,576 words). The main operational memory and main permanent memory are directly connected to the system processors, and the main operational and large operational memories have two-way exchange with a speed of roughly 20 million bytes per second in each direction, performed simultaneously with computations in the processors of the lines. External exchange is carried on through both the main operational and large operational memory units. All three internal memory devices are a single memory field from the standpoint of the user. The entire internal memory has uniform virtual addressing. Addressing is done with a precision down to the half-word; the actuating address contains 22 bits.

Thirty bit positions in the command are allocated for the address; four bits contain the number of the register for basing, four contain the number of the register for indexing, and 22 are used for displacement. There are 16 special registers that perform the functions of base and index registers. They are connected to memory and to the special processes to perform index operations. Formation of the actuating address for access to memory is organized by analogy with the IBM 360/85 system and the IBM 370 family [202]. Formation of the mathematical address is controlled by the user, while the actuating address is controlled by the operations system with the help of descriptors.

The system has a broad, variable format for access to main memory: 2-64 bytes of information may be extracted in one query.

The format of the commands that are interpretable by the central control unit is variable: from four to 24 bytes. During the execution of a full-format command, in one cycle of the system the following are performed: one operation of the control unit; two arithmetic-logical operations in two lines of the main processors; two references to main memory for operands at different addresses; and, one more reference for the next command (and the immediate operand). In addition, information arrays can be exchanged in this way with other M-10 systems.

External exchange is carried on through a multiplex channel which provides a total carrying capacity of about seven megabytes a second and has 24 duplex subchannels. Up to six units of one type, such as terminals with typewriters and punched tape equipment, alphanumeric printers, punched card equipment, and a keyboard on an engineering console to keep a hardware log, may be connected to each subchannel. Peripheral YeS [Unified System] devices such as hatchured displays with keyboards and light pencils or magnetic disks and tapes can also be connected to the channel through additional interlinking units.

Three basic parallel processes can be identified in the M-10 system: (1) computations in the central parts, which are in turn, parallel synchronous processes; (2) exchange of information between the main and large operational memory units; (3) external exchange through the multiplex channel. The processes of monitoring (by special circuits) the working condition of the equipment and monitoring user programs (for example, whether they have privileged operations) can be added to these processes. The interaction of these independent and simultaneous processes is accomplished through a multilevel program interrupt system whose free input may receive up to 32 external signals.

The operations system of the M-10 computer system provides the following: user dialog in a time-sharing mode; access to translators and program debugging means in algorithmic languages; reference to peripheral equipment on the logical level; reference to standard procedures; and, use of standard programs of the library as ready-loading modules. The library also includes model programs of linear algebra, approximation of functions, quadratures, integration of conventional differential equations, equations in partial derivatives, and others.

The resident parts of the operations system are stored in the main permanent memory. During the work process the operations system and the automatic exchange device move the necessary segments from the large memory to the main operational memory. When working in the time-sharing mode each user is assigned 20-80 microseconds, which insures balance in the characteristics of the main processors and the main and large memories [198].

The calculated productivity of the system is assured when solving problems with natural parallelism that usually require performance of operations on multidimensional vectors or functions, given by values in sets of discrete values of variables. When solving problems of this sort which require large memory volume, the actual productivity may significantly exceed calculated productivity.

The primary logical circuits of the M-10 systems are built with microcircuits. Both operational memory units are constructed with ferrite cores. The data medium in the permanent memory is replaceable metallic punched cards; it is a condenser-type memory. The first industrial models of the M-10 system demonstrated high performance characteristics.

The M-10 system provides for circuits that make it possible to join up to seven systems into a synchronous complex with a common cycle generator and virtual addressing of the system in the complex. In each work cycle the system can output an array of 64 bytes on its output lines and receive an array of the same size from any other system of the complex. No such complexes have been built as yet, but the communications registers of the computing systems are used as supplementary high-speed memory.

The M-10 system can be classified as an OKMDS/Vs [expansion unknown] with vector data flow (see Section 2.1) if we disregard the possibility of using two independent op codes in two lines.

Bibliographic Note. The general conception of the M-10 is described in [196, 197, 881], and the M-10 system itself is described in the article referred to in the footnote at the beginning of this section. Some questions of software and programming for systems of this kind (translation, standard programs, and paralleling) are considered in [33, 425, and 526].

FOOTNOTES

33. Belyakov, M. I., and Natanson, L. G., "Metalanguage, the Scheme of Translation, and Syntactic Analysis in the System of Constructing High-Performance Translators," PROGRAMMIROVANIYE 1975, No 1, pp 40-47.

196. Kartsev, M. A., "Questions of the Construction of Multiprocessor Computer Systems," VOPROSY RADIOELEKTRONIKI, SERIYA ELEKTRONNAYA VYCHISLITEL'NAYA TEKHNIKA 1970, Vyp 5-6, pp 3-19.

197. Kartsev, M. A., "Struktury Vychislitel'nykh Sistem i Ikh Effektivnost' pri Reshenii Raznykh Klassov Zadach" [Structure of Computer Systems and Their Efficiency in Solving Various Classes of Problems], Moscow, ITM i VT AN SSSR, 1977, Preprint No 11, 14 pages.

198. Kartsev, M. A., "Arkhitektura Tsifrovykh Vychislitel'nykh Mashin" [The Architecture of Digital Computers], Moscow, Nauka, Glavnaya Redaktsiya Fiziko-Matematicheskoy Literatury, 1978.

202. Kattsan, G., "Vychislitel'nyye Mashiny Sistemy 370" [System 370 Computers], Moscow, Mir, 1974, 508 pages (translated from English).

425. Shavlovskaya, S. A., "The Library of Standard Programs for a Multiprocessor," VOPROSY RADIOELEKTRONIKI, SERIYA ELEKTRONNAYA VYCHISLITEL'NAYA TEKHNIKA 1970, Vyp 9.

426. Shavlovskaya, S. A., "Paralleling in the Computation of Defined Integrals," VOPROSY RADIOELEKTRONIKI, SERIYA ELEKTRONNAYA VYCHISLITEL'NAYA TEKHNIKA 1973, Vyp 7, pp 3-10.

881. Kartsev, M. A., "On the Structure of Multiprocessor Systems," PROC. IFIP CONGRESS 71, Ljubljana, 1971, Amsterdam, North Holland Publ. Co., 1972, pp 559-564.

[146-11,176]

11,176
CSO: 1863

UDC 681.32-181.48(47+57)

MICROCOMPUTERS BASED ON n-CHANNEL METAL OXIDE SEMICONDUCTOR LARGE-SCALE INTEGRATED CIRCUITS (MOS LSIC's) DESCRIBED

Moscow MIKRO-EVM 'ELEKTRONIKA S5' I IKH PRIMENENIYE in Russian 1980 (signed to press 3 Nov 80) pp 64-72

[Chapter 3 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", 35,000 copies, 160 pages]

[Text] Chapter 3. Microcomputers Based on n-Channel MOS LSIC's

3.1. Basic Set of n-Channel MOS LSIC's

The element base of the "Elektronika S5-21" single-board microcomputer and of microprocessor functional modules is a set of n-channel LSIC's. LSIC's fabricated according to the n-channel technology are characterized by higher speed and degree of integration of elements on a chip, as compared with p-channel MOS LSIC's.

Heightening of the degree of integration of elements on a chip has made it possible to increase considerably the functional capacity of LSIC's and at the same time to reduce considerably the minimum set of LSIC's required for constructing microcomputers.

The following are the key characteristics of n-MOS LSIC's: number of transistors on a chip for irregular structures--8000 maximum, for regular structures--20,000 maximum; supply voltage--+5 V $\pm$ 5 percent, +12 V $\pm$ 5 percent and -5 $\pm$ 5 percent; clock frequency--2 MHz; levels of input and output signals conform to levels of TTL [transistor-transistor logic] circuits.

n-LSIC chips are mounted in ceramic packages with 48 terminals and in glassy alloy packages with 24 terminals. The composition of a set of MOS LSIC's is presented in table 3.1.

Table 3.1.

Designation of LSIC	Purpose
K586IK1	16-bit microprocessor
K586RU1	RAM [random access memory]
K586RYe1	ROM [read-only memory]
K586IK2	Multifunctional digital data I/O [input/ output] unit

Single-Chip Multiprocessor (K586IK1)

The single-chip 16-bit microprocessor (OMP) executes the instructions of the "Elektronika S5" instruction set. In order to perform the same functions, in the previous generation of computers the microprocessor contained 11 p-LSIC's. Another advantage of the OMP is its high speed, which is 20 times faster than that of a p-channel microprocessor.

Let us note one more property of an OMP. As a rule, the changeover from a multichip to a single-chip multiprocessor is accompanied by the rejection of microprogramming and the use of a control logic based on a programmed logic array (PLA). Furthermore, the possibility is eliminated of the processor's emulating certain additional functions for more flexible utilization of the computer. In order to avoid this, in developing the K586IK1 OMP a solution was implemented which makes it possible to create a microprocessor based on a PLA and possessing the property of so-called external microprogramming. In other words, in addition to macroinstructions from the "Elektronika S5" instruction set structure, 16-bit microinstructions can enter the OMP's information input lines. Thus, there is the possibility of using microprograms for performing additional functions not provided by the instruction set. A structural diagram of the OMP is presented in fig 3.1. It includes the following: a 16-bit parallel ALU [arithmetical unit]; a control unit based on a PLA; 16 16-bit general-purpose program registers; an instruction register; an address register; an information register; a data, address and control signal line; and additional circuits for comparison, lockout, and for forming synchronization pulses.

The external interface of the single-chip microprocessor is a bidirectional combined address and information line and a control signal line. Combining the address and information lines is a means of reducing the amount of terminals and of economizing on the area of the OMP's LSIC chip (by reducing the number of buffer circuits). A line of this sort makes it possible also to utilize economically the area of the printed circuit board of a microcomputer, which is especially important in creating single-board configurations for computers. With this, of course, it is necessary to take into account some loss in speed in microcomputers.

The microprocessor can operate in real time and in the multiprogram mode, and also make possible the mode of direct access to the microcomputer's memory. The following are the key specifications of OMP LSIC's of the K586IK1 type: word length--16 bits; address--15 bits; direct addressing of 32,768 words; number of key instructions--31; speed--20,000 operations per second of the "register-register"

type; clock frequency--2 MHz; package--ceramic with 48 terminals; supply voltage--+15 V and +12 V; and power requirement 900 mW.

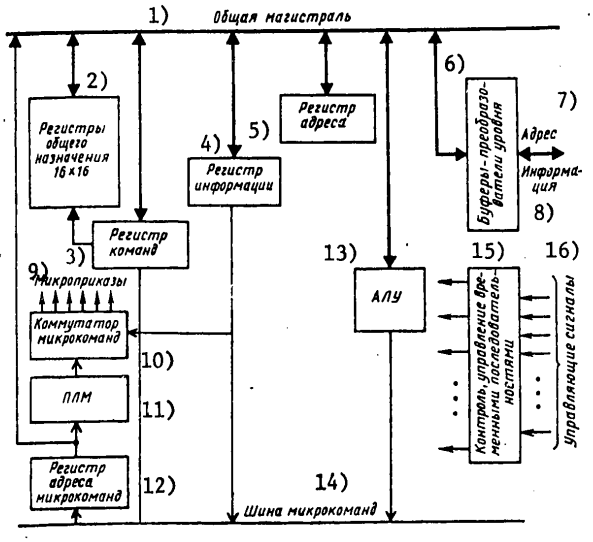


Figure 3.1. Structural Diagram of Single-Chip 16-Bit Microprocessor

Key:

1. Common line
2. 16 X 16 general-purpose registers
3. Instruction register
4. Information register
5. Address register
6. Buffers - level converters
7. Address
8. Information
9. Microinstructions
10. Microinstruction distributor
11. PLA
12. Microinstruction address register
13. ALU
14. Microinstruction line
15. Time sequence monitor and control
16. Control signals

Memory LSIC's (K586RU1 and K586RYe1)

The K586RU1 RAM microcircuit is executed on the basis of six transistor static storage elements. The information capacity of RAM LSIC's is 1024 bits and has a 256 X 4 organization. All four number inputs and outputs are combined. The output buffer circuits have three states, whereby after the arrival of an inhibit signal the output assumes the high-impedance state. The RAM access cycle occupies four clock pulses (2 μs) and information is transferred at the end of the third clock period.

The power dissipated by the RAM LSIC is 150 mW. The package is a glassy alloy one with 24 terminals.

The K586RYe1 ROM has an information capacity of 16K bits and a 1024 X 16 organization. Address inputs are combined with information outputs. As in the RAM, information is read out at the end of the third clock pulse. The power dissipated by the ROM LSIC is 240 mW. The package is a ceramic one with 48 terminals.

The RAM and ROM microcircuits have been unified to the maximum. Their components, such as the address register, synchronizer, output buffer stages, etc., have been made identical. It should be noted that a number of structural and circuitry decisions forming the basis for the development of the RAM and ROM LSIC's, such as the combining of address and information lines with time sharing of signals, the 16-bit structure of the ROM and the four-bit structure of the RAM, and the employment of a working storage cell of the static type, were aimed at reducing the printed circuit board area of a single-board microcomputer and of functional modules with an internal storage arrangment.

Digital Input/Output LSIC (K586IK2)

This has a rearrangeable structure and is a multifunctional device. The TsVV [digital I/O] LSIC makes possible the following: the exchange of parallel codes through two eight-bit channels, the exchange of serial codes through a single eight-bit channel (or the organization of a program controlled timing mode) and the processing of interrupt signals for eight inputs, thereby occupying eight-bit parallel information I/O channels. A diagram of a UVV [I/O device] LSIC (K586IK2) is presented in fig 3.2.

Channels for exchanging eight-bit parallel codes can be adjusted to receive (read out) information or to process interrupt signals. In the latter case a mask is entered into one of the channels via the software, and external interrupt signals enter the other.

The serial I/O channel can operate both as an eight-bit counter (for adding or subtracting input pulses with a frequency up to 300 kHz) or as a shift register (shifting information to the left or to the right in relation to external clock pulses with a frequency of up to 600 kHz), whereby in both modes can be formed a signal for equality of the counter - shift register's contents with the contents of the program controlled control point register.

3.2. "Elektronika S5-21" Microcomputer

The creation of the "Elektronika S5-21" microcomputer on the basis of the set of n-MOS LSIC's described above represented a further development of single-board models of microcomputers for the purpose of increasing capacity and reducing cost. This computer is program-compatible with the "Elektronika S5" series of microcomputers and from the viewpoint of the programmer-user represents a 16-bit computer with an address field of up to 32K 16-bit words. This address field includes also the addresses of I/O units, which are accessed by means of the same operations as for access to the memory [9].

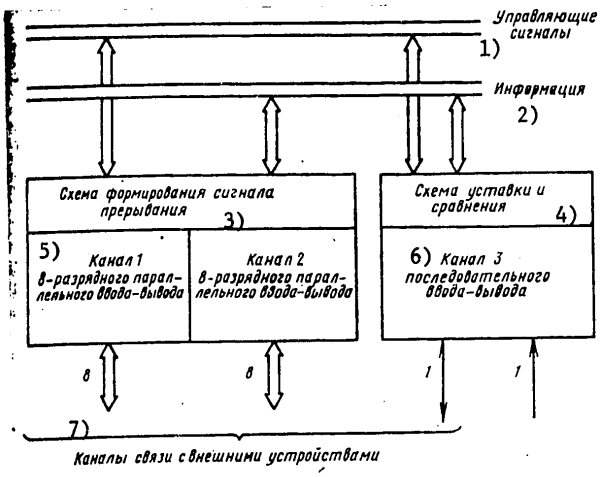


Figure 3.2. Structural Diagram of K586IK2 LSIC

Key:

1. Control signals
2. Information
3. Circuit for forming interrupt signal
4. Control point and comparison circuit
5. 8-bit parallel I/O channel 1 and 2
6. Serial I/O channel 3
7. Channels for coupling with peripherals

A structural diagram of the "Elektronika S5-21" microcomputer is presented in fig 3.3. The bidirectional combined address and information line couples the microprocessor with storage and I/O units placed on the board. The RAM with a capacity of 256 16-bit words is executed with four K586RU1 LSIC's. The ROM with a capacity of 2048 16-bit words is executed with two K586RYe1 LSIC's.

The I/O unit is executed with four type K586IK2 LSIC's and makes possible the following: the reception from peripherals of eight-bit pulse, synchronous, asynchronous or potential data through one to eight channels, the transmission to external lines of eight-bit potential or pulse information through one to eight channels (the total number of eight-bit I/O channels is eight), the conversion of a serial eight-bit code into a parallel and vice-versa through one to four channels with a clock frequency of not less than 600 Hz, division of the input frequency over each of four channels with a program-changeable factor from 1 to 256 with the possibility of reversal with a frequency of not less than 300 Hz, counting of an assigned number of pulses within the range of from 1 to 256 with a

frequency of not less than 300 kHz with the formation of a signal for equality to the assigned number for each of four serial channels (the total number of eight-bit channels for the input/output of serial codes and for processing frequency signals is four), the formation of a mesh of quartz-controlled frequencies of 1200, 600, 120, 15, 7.5, and 1 kHz and 100, 10 and 1 Hz, and the reception of interrupt signals through eight priority channels [12].

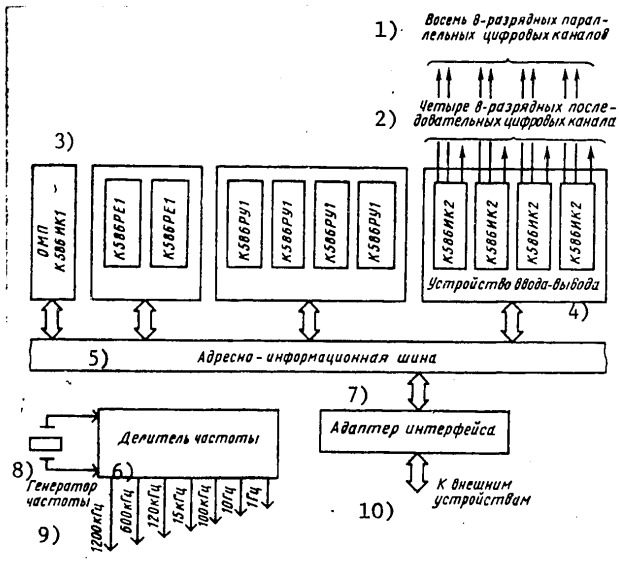


Figure 3.3. Structural Diagram of "Elektronika S5-21" Microcomputer

Key:

1. Eight 8-bit parallel digital channels
2. Four 8-bit serial digital channels
3. K586IK1 OMP
4. I/O unit
5. Address and information line
6. Frequency divider
7. Interface adapter
8. Clock
9. 1200 kHz
10. To peripherals

Structurally these microcomputers are in the form of a printed circuit board with components mounted on one side. At two opposite ends are installed two type GRPM-61 connectors each. The printed circuit board is attached to a frame to which covers are attached. Openings are provided in the computer's structure for the purpose of attaching guides which are components of user's systems. The external appearance of the "Elektronika S5-21" microcomputer is shown in fig 3.4 [photograph not reproduced].

The following are the key specifications of the "Elektronika S5-21" microcomputer: word length--16 bits; number of basic instructions--31; capacity of internal RAM--512 bytes; capacity of internal ROM--4096 bytes; addressing capability of up to 64K bytes; clock frequency--2 MHz; 64 parallel digital channels; four serial digital channels; input and output levels compatible with levels of TTL circuits; supply voltage +5 V $\pm$ 5 percent and +12 V $\pm$ 5 percent; power consumption--20 VA; overall dimensions--309 X 252 X 29 mm; weight--1.2 kg; operating temperature-- -10 to +50 °C.

3.3. Microprocessor Functional Modules for "Elektronika S5-21" Microcomputers

The set of microprocessor functional modules based on n-MOS LSIC's is designed to broaden the utilization capabilities and to improve the capacity of systems based on "Elektronika S5-21" microcomputers [9]. The composition of the set of modules is presented in table 3.2.

Table 3.2.

Designation of module	Purpose of module
"Elektronika S5-2101"	Module for coupling microcomputers with transmitters and receivers of analog signals (d.c. voltage)
"Elektronika S5-2102"	Module for coupling microcomputers with transmitters and receivers of digital signals
"Elektronika S5-2103"	Module for coupling with terminal equipment and punched tape I/O units
"Elektronika S5-2105"	RAM module
"Elektronika S5-2106"	Display adapter module
"Elektronika S5-2107"	Program debugging unit module
"Elektronika S5-2108"	ROM module

The structural design of all MFM's [microprocessor functional modules] is the same as for "Elektronika S5-21" microcomputers (overall dimensions--309 X 252 X 29 mm).

"Elektronika S5-2101" MFM

This module makes possible distribution over 32 channels and the analog-digital conversion of d.c. signals varying over the range from -5 to +5 V; the access and conversion time for a single channel is not greater than 200 μs and the distribution and conversion error is not greater than 0.2 percent.

"Elektronika S5-2102" MFM

This module makes it possible to couple "Elektronika S5-21" microcomputers with peripherals through digital I/O units, it contains six type K586IK2 LSIC's and has the following:

Digital outputs (eight bytes) designed for the connection of TTL circuits with a load current of up to 16 mA.

Digital inputs or outputs (four bytes) designed for connection of TTL circuits with a load current through output channels of up to 50 mA (adjustment for input or output is accomplished by appropriate switching in the module's external connector).

Serial input/output (six channels). Each channel when appropriate switching is performed in the output connector of the module can operate as an eight-bit counter with an input frequency of up to 300 kHz (for adding or subtracting) or as an eight-bit reversible shift register with a shift clock frequency of up to 600 kHz. Each serial channel in any operating mode issues a signal for equality of the contents of the shift register - counter and the contents of the program controlled control point register.

"Elektronika S5-2103" MFM

This module makes it possible to couple the "Elektronika S5-21" microcomputer with the following units: an FS-1501 or SP-3 facsimile unit, a PL-150 or PL-80 puncher, an RTA-6, RTA-7, RTA-60 and T-63 teletype, and others operating in MKT-2 code with a transmission speed of up to 100 bauds.

"Elektronika S5-2105" MFM

This is a RAM with a capacity of 16K 16-bit words, designed on the basis of a K535RU3 LSIC (16K bits), and contains a hardware-controlled restoration circuit.

"Elektronika S5-2106" MFM

This module makes it possible to couple with a VKU [video monitor] (based on a CRT [cathode ray tube]) for the purpose of forming on the CRT's screen alphanumeric information with a capacity of up to 1024 characters with a program controlled format. The number of lines is 8, 16 and 24, the number of characters in a line is 16, 32 and 64, and the input language is KLI-7 with a 128-character alphabet.

This MFM has an editing function: characters are entered in reference to a controlled tag; it is possible to shift characters in a line and to shift lines up and down.

"Elektronika S5-2107" MFM

This module is a console containing a keyboard and displays and together with the "Elektronika S5-21" microcomputer, containing a program for processing signals from

the console, makes it possible to implement all console algorithms for operation of the microcomputer.

"Elektronika S5-2108" MFM

This is an ROM with a capacity of 4K 16-bit words and is designed on the basis of KR55RT5 LSIC's having a 512 X 8 structure.

[136-8831]

8831
CSO: 1863

MICROELECTRONIC ANALOG PROCESSOR FOR MATRIX READING OF IMAGES

Moscow MIKROELEKTRONIKA in Russian Vol 10, No 1, Jan-Feb 81 pp 87-94

[Article by S. V. Svechnikov and M. A. Popov]

[Text] Digital and analog techniques, based on a general purpose or special computer, are being used more and more for optical image processing and recognition.

When a general purpose computer is used an optical image is converted with a scanner and single-channel photoelectric transducer to an electrical signal, digital samples of which are fed serially into the computer for future digital processing. The advantages of using a general purpose computer are the ability to use a very broad class of processing and recognition algorithms, and high precision [1, 2]. However, the sequential nature of the operation of most modern general purpose computers and their transmission capacity (speed) are not sufficient for the processing and recognition of images in real time [3].

In this connection preference is given to special computers for solving numerous image recognition and processing problems. The functioning of such computers can be based on different principles, and there are specialized computers both with digital, and with analog representation and processing of image signals [4-7].

We note that the main method of increasing the transmission capacity of a computer, irrespective of how signals are represented, is to parallelize processing and computing operations. For instance, the special digital computer described in [6] can parallel-process images as large as 96 × 96 elements.

In parallel computations image signals usually are fed into computer memory ahead of time, and they are processed by means of access to memory. This makes it possible to use traditional single-channel photoelectric transducers for reading and feeding images into a parallel computer, but it greatly hampers the utilization of the potential capabilities of such computers in terms of speed.

A microelectronic processor for parallel (matrix) reading and input of images into a parallel-computing computer is examined in this article. The processor reads the image of a multielement aperture, formed by a matrix photoreceiver. The current positions and sizes of the aperture, which is rectangular, are set with the aid of a control logic system, based on analog neurons.

An important feature of the controlled parallel reading technique used in the work is the feasibility of combining the image reading process with preprocessing, for example by dividing an image into individual fragments of different sizes.

The described properties of the processor make it possible not only to speed up image input, but in many cases also to shorten the time the computer needs to process the image.

Structure of Processor

In the main functional element of the processor is a neuron. The neuron used in this work is similar in terms of properties to the element examined earlier [9], but unlike the latter, it has a static characteristic that can be tuned in a wide range. A schematic electrical diagram of the neuron, built on the basis of an operations amplifier, and its conventional designation are shown in Figure 1. The neuron is used in the processor in four different functional circuits: aperture, key, control and threshold. To each circuit of the neuron corresponds a certain static characteristic (see the table).

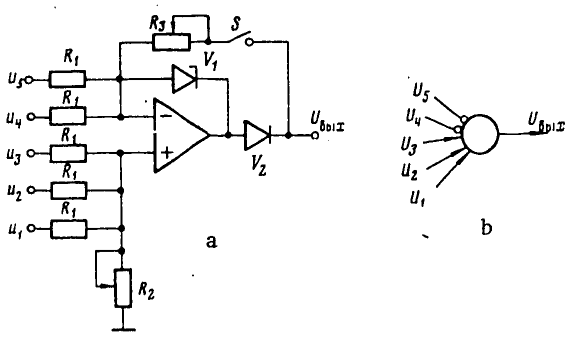


Figure 1. Electrical diagram of neuron (a) and its conventional symbol (b). Arrow indicates stimulating (noninverting) input, circle indicates retarding (inverting) or threshold input: 1 -- image being read; 2 -- lens; 3 -- photoreceiver matrix; 4 -- aperture neuron matrix; X and Y -- aperture neuron control channels in mutually orthogonal directions; in X channel: 5, 9 -- voltage dividers; 6 -- threshold neuron unit; 7 -- weight cell unit; 8 -- adder; 10 -- control neuron unit; 11 -- key neuron unit. Double connecting lines denote vector connections between units; u -- voltage. [ВЫХ=out]

A structural functional diagram of the processor is presented in Figure 2. The processor also contains a weight cell and a voltage divider. The weight cell is an amplifier with the corresponding gain (weight) k. A combination of weight cells in the X channel comprises a line of elements, the spatial distribution of the weights of which obeys the law

$$k_i=K(i-n)/(1-n), \quad i=1,2,\ldots,2n-1 \tag{1}$$

and which is shown in Figure 3a. It follows from expression (1) that $k \gtrless 0$, $|k| \leq K$. The analogous distribution is fixed in the Y channel.

Variations of Functional Connection of Neuron in Processor Units

1) Функциональный тип нейрона	2) Схема включения	3) Положение переключателя S	4) Статическая характеристика
5) Апертурный нейрон	θ_1=const; U_{x_i}, U_{y_i}, $U_{вых}$; от i j-фотоприемника 6)		$U_{вых}$; θ_1, ΣU
7) Пороговый нейрон	θ_2, U, $U_{вых}$		$U_{вых}$; θ_2, U
8) Управляющий нейрон	U_2, U_1, $U_{вых}$		$U_{вых}$; ΣU
9) Ключевой нейрон	U_3, U_2, U_1, $U_{вых}$		$U_{вых}$; θ_3, U_1; $\theta_3 = U_3 + U_2$

Key: 1. Functional type of neuron
2. Connection diagram
3. Position of switch S
4. Static characteristic
5. Aperture neuron
6. from i j-th photoreceiver
7. Threshold neuron
8. Control neuron
9. Key neuron
[вых=out]

The voltage divider consists of a chain of series-connected resistors. The distribution of the output voltages of divider 9 obeys the law

$$u_i^{(9)} = u_2(i-n)/(n-1), \quad i=1,2,\ldots,2n-1 \tag{2}$$

and is shown in Figure 3b. The distribution of the output voltages of divider 5 obeys the law

$$u_i^{(5)} = u_1(i-1)/2(n-1), \quad i=1,2,\ldots,2n-1 \tag{3}$$

and is shown in Figure 3c. The voltage dividers in the Y channel are of the analogous design.

The processor operates as follows (Figure 2). Image 1 to be read is projected in parallel by lens 2 onto photoreceiver matrix 3, where it is converted to a set of electrical signals. Each photoreceiver of the matrix has an independent output,

which is connected to the input of the corresponding aperture neuron matrix 4. The number of aperture neurons here is equal to the number of photoreceivers and the ij-th photoreceiver is connected to just the ij-th aperture neuron, i.e., a one to one correspondence exists between the elements of matrices 1 and 4. By virtue of this correspondence it is possible to accomplish controlled (selective) reading of the output signals of the matrix photoreceiver (and thus of individual fragments of the image), for which purpose a reading aperture is formed on the aperture neuron matrix. The reading aperture is rectangular (Figure 4) and the shape is determined unequivocally by the following parameters: the coordinates of the bottom left element $i_0 j_0$ and two linear dimensions $\Delta_i = i_1 + 1 - i_0$ and $\Delta_j = j_i + 1 - j_0$. The largest possible reading aperture obviously measures $\Delta_i \times \Delta_j = (2n - 1) \times (2m - 1)$.

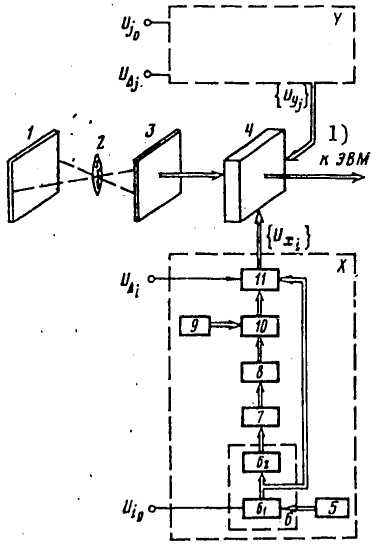


Figure 2. Structural functional diagram of microelectronic analog processor for matrix image reading.
Key: 1. To computer

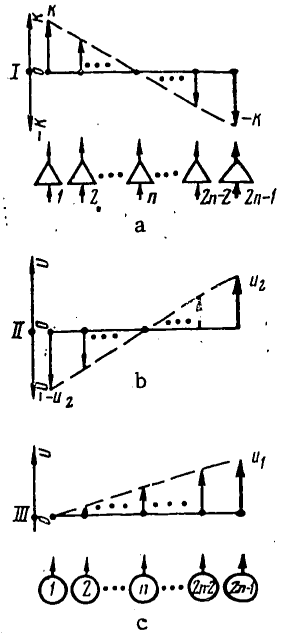


Figure 3. Diagrams of spatial distribution: of corresponding weights k (gains) of weight cells (a), of output voltages of divider 9 (b) and of divider 5 (c): I -- weight of B7; II -- outputs of B9; III -- outputs of B5.

The principle on which the reading aperture is formed is the following. Each aperture neuron operates in the threshold amplification mode (see the table), i.e., a neuron can either pass (amplify) the photoreceiver signal that stimulates its input, or not pass that signal. In the initial state the identical voltage $u = \theta_1$ is applied to all the aperture neurons through the threshold input. This voltage

is set above the maximum possible photoreceiver signal, and therefore all the output signals of the aperture neurons are zero in the initial state.

In order to read an image it is necessary to change at least some of the aperture neurons to the zero threshold state. Then these neurons will proportionately amplify the signals of the corresponding photoreceiver, and thus a reading aperture is formed.

The aperture neurons are controlled through two control channels X and Y. To the inputs of the X channel are fed external control signals

$$u_{i_0}=\alpha_0 i_0/2(n-1), \quad i_0=1,2,\ldots,2n-1, \tag{4}$$

$$u_{\Delta_i}=\alpha_\Delta \Delta_i/2(n-1), \quad \Delta_i=1,2,\ldots,2n-1, \tag{5}$$

where α_0 and α_Δ are constant coefficients. Signals (4) and (5) determine the position of the aperture and its linear dimension in direction X, respectively. To the inputs of the Y channel are fed external control signals

$$u_{j_0}=\alpha_0 j_0/2(m-1), \quad j_0=1,2,\ldots,2m-1, \tag{6}$$

$$u_{\Delta_j}=\alpha_\Delta \Delta_j/2(m-1), \quad \Delta_j=1,2,\ldots,2m-1, \tag{7}$$

which determine the position of the aperture and its linear dimension in direction Y, respectively. It follows from expressions (4)-(7) that the control signals are multilevel signals.

Depending on the magnitude of the control signals, vector signal $\{u_{x_i}; i = 1, 2, \ldots, 2n - 1\}$ is generated at the outputs of channel X, each element of which may acquire just one of two possible values: $u_{x_i} = 0$ or $u_{x_i} = 0.5\theta_1$. Likewise for channel Y $\{u_{y_j}, j = 1, 2, \ldots, 2m - 1\}$, and $u_{y_j} = 0$ or $u_{y_j} = 0.5\theta_1$.

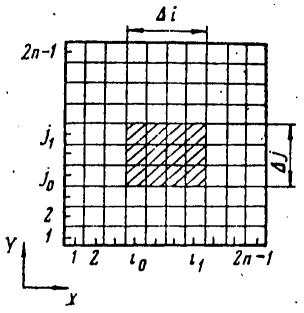


Figure 4. Image field and possible position and parameters of reading aperture (shaded area).

The outputs of the X and Y channels are connected to the stimulating inputs of the aperture neurons through a system of orthogonal bars, so that all 2m - 1 aperture neurons of the i-th column of the matrix are connected to the i-th output of the X channel, and all 2n - 1 aperture neurons of j-th line are connected to the j-th output of the Y channel.

To switch any aperture neuron to the zero threshold state it is necessary to apply simultaneously equal $0.5\theta_1$ signals to both its stimulating inputs, connected to the X and Y channels. Then the neuron passes (amplifies) the signal of the corresponding

photoreceiver, and the combination of these "photoreceiver-aperture neuron" pairs forms a reading aperture.

Let us examine how signals are generated in the X and Y channels, which determine the state of the aperture neurons, depending on the external control signal. Both channels are of the identical design; the only possible difference is in the number of elements in a layer. Therefore we will describe just one channel, the X channel, a detailed diagram of which is shown in Figure 5 (the same enumeration of the units and symbols is used as above). We assume that the inputs of the channel are acted upon by external control signals $u_{i_0}{}^*$ and $u_{\Delta_i}{}^*$, which determine, respectively, the photoreceiver number $i_0{}^*$ and aperture length $\Delta_i{}^*$ in direction X. Signal $u_{i_0}{}^*$, along with the voltages of divider 5, produces the summary input voltage distribution of neurons 6_1 of the first layer of unit 6, illustrated in Figure 6a. The reaction of the first layer of threshold neurons in consideration of their properties (see the table) is shown in Figure 6b, and the reaction of the second layer of threshold neurons 6_2 is illustrated in Figure 6c. It follows from Figure 6c that of all the output signals of the neurons of the first layer of unit 6, only one is not zero, and that is the output signal of the $i_0{}^*$-th neuron. This signal is "weighted" with weight $k_{i_0}{}^*$, and, passing through adder 8, is applied to the first stimulating input of all the neurons of unit 10. To the second stimulating input of these neurons is applied the voltage from the corresponding tap of divider 9. A diagram of the summary stimulus at the input of the neurons of unit 10 is shown in Figure 6d.

A neuron, as follows from its static characteristic, is sensitive to just positive input signals, and therefore the distribution of the output voltages of the control neurons of unit 10 will acquire the form shown in Figure 6e.

The output signal of each of the control neurons goes to the retarding input of the corresponding key neuron of unit 11. The key neurons are connected through the other retarding input to the outputs of neurons 6_1, and the retarding input of the i-th neuron of unit 11 is connected to the output of the (i + 1)-st neuron 6_1. As a result of this connection, as is shown in Figure 6f, the first ($i_0{}^*$ - 1) key neurons have the highest threshold, the $i_0{}^*$-th key neuron as the zero threshold, and the thresholds of the other key neurons increase linearly with the number of neuron in the unit.

External control signal $u_{\Delta_i}{}^*$, which determines the length of the aperture in direction X, goes to the stimulating input of the key neurons. For the above-examined distribution of thresholds, of all the key neurons, those with the numbers from $i_0{}^*$ to $i_1{}^*$, inclusively, are activated, and the number of these neurons depends on the size of applied signal $u_{\Delta_i}{}^*$ and is calculated in accordance with expression (5). We note that in order to create a single-element aperture,

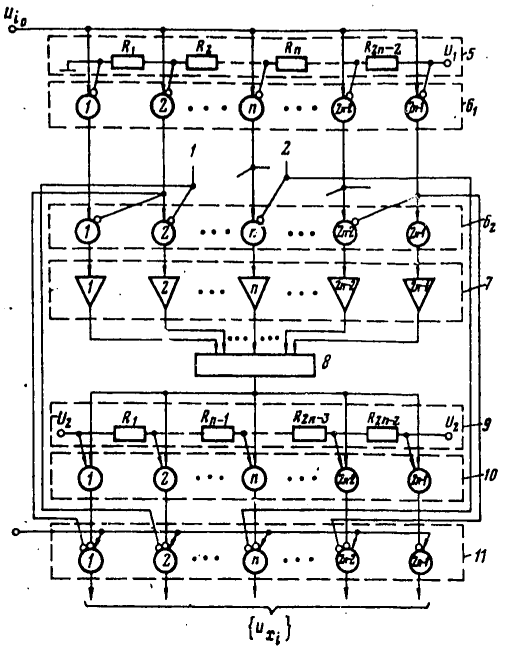


Figure 5. Detailed diagram of X channel, controlling aperture neurons: 1 -- from third neuron; 2 -- from (n + 1)-st neuron.

as follows from expression (5), it is necessary also to apply the corresponding control signal, since when $u_{\Delta_i}^* = 0$ activation does not occur.

After activating the key neurons form the spatial output signal distribution pattern $\{u_{x_i}\}$ in channel X, shown in Figure 6g, and the size of the output signal of each of the activated neurons is $0.5\theta_1$, and the output signals of the other key neurons are zero.

The spatial distribution pattern of the Y channel output signals is formed in the same way.

The output signals of the X and Y channels, being applied to the aperture neuron matrix, form a reading aperture of size $\Delta_i^* \times \Delta_j^*$ in the corresponding region of the image field. With the aid of this aperture the signals of the corresponding image fragment are fed simultaneously into a parallel computer.

Application of Processor

The characteristic features of the examined processor are the use of a matrix photoreceiver, high homogeneity (basically the same type of microelectronic neurons are used), the parallel structure and layer by layer execution of logic

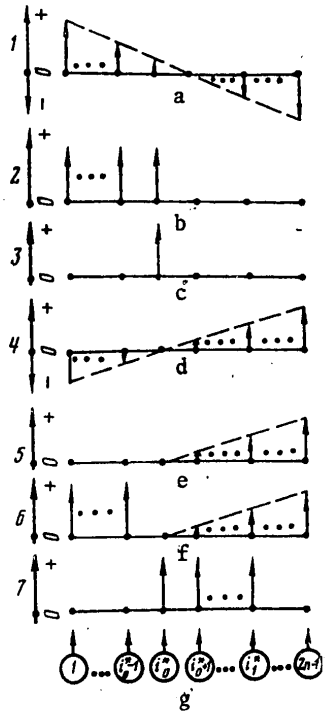


Figure 6. Diagrams of spatial distribution of signals in neuron units of X channel for case examined in text: 1 -- input $B6_1$; 2 -- output $B6_1$; 3 -- output $dB6_2$; 4 -- input B10; 5 -- output B10; 6 -- threshold input B11; 7 -- output B11 $\{U_{xi}\}$.

operations. These properties of the processor not only help to improve the speed and reliability of the reading and input of images into a parallel computer, but also to take advantage of a number of additional opportunities. These additional opportunities consist primarily in the feasibility of combining the image reading process with preprocessing, in particular by dividing an image into fragments by means of successive assignment of the corresponding levels of the external control signals.

Another possibility is the use of functional transformation (for example expansion into a spectrum) of an image fragment within the aperture.

We mention in conclusion that the processor also can be used as an external system of general purpose computers. The optical signal storage mode, in which the

matrix photoreceiver operates, provides a high signal to noise ratio during image input into a computer.

BIBLIOGRAPHY

1. Yaroslavskiy, L. P., "Vvedeniye v tsifrovuyu obrabotku izobrazheniy" [Introduction to Digital Image Processing], Moscow, Sovetskoye radio, 1979.

2. "Obrabotka izobrazheniy i tsifrovaya fil'tratsiya" [Image Processing and Digital Filtering], edited by T. Khuanga, Moscow, Mir, 1979.

3. Krasikov, V. A. and M. V. Khatuntseva, "Aerospace Research," "Obrabotka videoinformatsii na EVM" [Video Information Processing by Computer], Moscow, Nauka, 1978, p 213.

4. Stamopoulos, C. D., IEEE Trans. Comput., 1975, Vol C-24, No 4, p 424.

5. Svechnikov, S. V., M. A. Popov, A. M. Shkvar and D. I. Chus', MIKROELEKTRONIKA [Microelectronics], 1979, Vol 8, No 4, p 362.

6. Cordella, L. P., M. J. B. Duff and S. Levialdi, IEEE Trans. Comput., 1978, Vol C-27, No 10, p 904.

7. Preston, K., "Sb. Raspoznavaniye obrazov pri pomoshchi TsVM" [Collection: Pattern Recognition with Digital Computer], Moscow, Mir, 1974, p 143.

8. "Skaniruyushchiye ustroystva na ELT vysokogo razresheniya (dlya EVM)" [High-Resolution CRT Scanning Systems (for Computers)], edited by S. T. Vas'kov, Novosibirsk, Nauka, 1978.

9. Svechnikov, S. V., M. A. Popov and A. M. Shkvar, MIKROELEKTRONIKA, 1977, Vol 6, No 5, p 388.

7872
CSO: 1863

BASIC CHARACTERISTICS OF YES SYSTEM MAGNETIC DISK STORAGE UNITS

Moscow EVM I VYCHISLITEL'NYYE SETI in Russian 1980 (signed to press 20 May 80) p 220

[Table from book "Computers and Computer Networks" by Vasiliy Nikolayevich Kruishin, Inna Nikolayevna Buravtseva, Nina Mikhaylovna Pushkina, and Nina Grigor'yevna Chernyak, Izdatel'stvo "Statistika", 18,000 copies, 328 pages]

[Excerpt] Table 8.5. Basic characteristics of magnetic disk storage units of the YeS [Unified System of Computers].

Technical Characteristics	YeS-5050 USSR	YeS-5051 USSR	YeS-5052 Bulgaria	YeS-5055 GDR	YeS-5056 Czechoslo.
Type of Accumulator	Replaceable	Permanent	Replaceable	Replaceable	Replaceable
Capacity, Mbytes	7.25	125	7.25	7.25	7.25
Number of Working Cylinders	200	384	200	200	200
Number of Tracks in a Cylinder	10	1	10	10	10
Number of Connectable Channels	2	2	2	1	2
Average Access Time, μsec	90	250	156	156	156
Speed of Data Transmission, Kbytes/sec	156	83.25	156	156	156
Code of Control Unit	YeS-5551	YeS-5551	YeS-5552	YeS-5555	YeS-5551

[Table continued, next page]

[Table 8.5 continued]

Technical Characteristics	YeS-5058 Czechoslovakia	YeS-5061 Bulgaria	YeS-5060 Hungary	YeS-5066 Bulgaria
Type of Accumulator	Replaceable	Replaceable	Permanent	Replaceable
Capacity, Mbytes	7.25	29.17	0.8	100
Number of Working Cylinders	200	200	256	404
Number of Tracks in a Cylinder	10	20	1	19
Number of Connectable Channels	1	2	-	2
Average Access Time, μsec	110	50	10	32
Speed of Data Transmission, Kbytes/sec	156	312	150	806
Code of Control Unit	YeS-5558	YeS-5561	-	YeS-5566

[116-11,176]

11,176
CSO: 1863

UDC 681.3

ORGANIZATIONAL ANALYSIS OF COMMON COMPUTER RESOURCES IN HOMOGENEOUS MULTIPROCESSOR COMPUTER SYSTEMS

Moscow AVTOMATIKA I TELEMEKHANIKA in Russian No 2, Feb 81 pp 164-174 manuscript received 23 Jan 80

ZABOLOTNYY, A. A.

[Abstract] Optimization of the processing system of multiprocessor computer systems by the lowest cost criterion for a given productivity is solved in the article. The number of computer systems, the structure of the multichannel switch and the traffic capacity of each communications channel are used as variable parameters for a homogeneous multiprocessor computer system with shared computer resources. The steady state instruction processing mode is examined and edge effects in mode switching are ignored. The throughput of a communications channel in relation to its cost is assumed to be given. Trunk line, radial and loop switching systems are selected for the study. Trunk line and radial switches are not suitable for a system with many modules, and local switching of shared computer resources is preferable. Figures 1; references 3: 2 Russian, 1 Western.
[127-7872]

SOFTWARE

UDC 681.3.06.001.2:681.3

PROBLEMS OF DEVELOPING AUTOMATED DESIGN SYSTEMS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 13-16

[Article by Candidate of Technical Sciences Ye. I. Artamonov]

[Text] Development of automated design systems (SAPR) is one of the most important state tasks which largely determine production efficiency and product quality. Automation of design is transformed from the "popular" theoretical field to an important constituent for increasing labor productivity, for example, in such sectors as aviation and motor vehicle building, the electronics industry, radio engineering, construction and architecture. SAPR are used extensively in creating complex parts, design of large integrated circuits (BIS), processing of geophysical information and constructing geological profiles and so on.

Development of SAPR is a very laborious and prolonged process and is possible only because of a large collective of developers. Therefore, combining the efforts of the leading organizations that design systems in different sectors of industry becomes a timely problem.

The Institute of Control Problems, USSR Academy of Sciences and Ministry of Instrument Making, Means of Automation and Control Systems, several years ago adopted measures to combine the efforts of a large number of organizations to develop the base of wide-use SAPR. The base for development was the Grafika system developed earlier at the Institute of Control Problems. The creative cooperation of the institute and coexecutor organizations was completely justified and is yielding the first results.

Some problems related to the hardware and software (MO) which occur during development of SAPR are considered in the article and possible ways of solving them are indicated using the Grafika system as an example.

The Hardware of SAPR

Hardware can be divided into three main groups, distinguished by the capacity of computer devices: medium-capacity computers, small computers and intellectual terminals. Computer devices may in the general case be combined into a network of various configuration which utilize, for example, medium-capacity computers at the upper level, small computers at the medium level and intellectual terminals at the lower level. The use of several levels of hardware makes it possible to organize collective work of users, to solve problems of different complexity, beginning, for

example, from simple problems of editing and conversion of drawings or alphanumeric information at the lower level, storage of standard graphical images, solution of individual calculating problems at the medium level and ending with organization of information retrieval systems and solution of problems of modelling complex processes at the upper level.

Thus, the general Motors Company (United States) utilizes a medium-capacity computer operating with several Applicon systems [1] constructed on the base of a small computer. The Applicon system is used to finish drawings of parts and to prepare control information for machine tools with ChPU [Numerical program control]. The automotive concern of the Leyland Gars Group in England [2] uses a dual IBM 370/158 computer and automatic Kongsberg drawing machine for its own purposes.

Systems which include medium- and low-capacity computers operate at a number of computer centers [3, 4]. The appearance of microprocessors made it possible to significantly expand the functional capabilities of terminals. The use of networks of these terminals facilitates the solution of many practical problems with minimum expenditures [5].

Each group of computer devices can be equipped with a wide array of external devices which provide various information input and output modes in the SAPR. The input information can be entered directly into the system from the keyboard of a console typewriter and the keyboard of alphanumeric and graphical displays in terms of a special input language. Information can also be entered from traditional input devices such as punch card, punch tape, magnetic disk and tape readers.

Devices for moving a special marker on the graphic display screen (a light pen, functional keyboard, sensory panels and so on) permit one to select the required data from the list displayed on the screen, to control the position of individual elements and the elements themselves and to edit the images. Automatic and semiautomatic drawing input machines permit acceleration of entry of preliminarily produced drawings and projections of an object for constructing three-dimensional drawings, level lines for subsequent plotting of terrain relief and so on.

The information in SAPR is retrieved to such devices as card and tape perforators, magnetic disks and tapes, printers, graph plotters, coordinate graphs, cathode-ray and light beam tubes [6], machines with ChPU and alphanumeric and graphic displays.

Vector and raster displays and also plasma panels are known among operating graphical information display devices. The appearance of matrix type printers made it possible to retrieve drawings at a higher rate than on line graph plotters, however, this led to significant complication of the software (PO), the function of which includes scanning the image pre-formed in the computer memory.

SAPR may contain a diverse set of external devices as a function of the capabilities of the user and the designation. In this case, computer entry from a teletype keyboard in terms of an input language, printout to a graph plotter, matrix type printer or intermediate carriers for subsequent control of production automatons can be used in the minimum configuration. The initial configuration of hardware is almost always subsequently expanded in a significant manner.

Thus, a wide range of computer devices distinguished by the capacity of the memory, speed and capabilities of the operating systems, is used in SAPR. The external devices have different accuracy, methods of encoding and forms of information display. These factors significantly affect the structure of the software and selection of the basic programming language.

The Software of SAPR

Software solves a wide range of problems occurring at all stages of design, beginning with postulation of the problem, synthesis and analysis and ending with the contract design and technological preparation for production.

The software of modern design systems includes two main components: the first contains means of describing graphical images, storage of libraries of standard drawings, editing, conversion, control of external devices and so on. These devices are usually called machine graphics devices. They significantly affect the efficiency of SAPR and determine their structure, universality and viability. The second component contains the means of solving applied problems toward which the SAPR is oriented.

The software for these components may in turn be divided into two groups: program packs and systems. Program packs should obviously be the basis of any systems exactly the same as the functional assemblies of the component base are the basis for construction of computer devices.

Packs for synthesis and analysis of control systems, solving problems of machine graphics [7], performing work at the contract design stage and so on are now known. Organization of software in the form of packs when solving specific problems requires that the user combine a set of subroutines by means of the operating system, which places definite restrictions on the level of the user's occupational training and makes an interactive operating mode difficult.

Development of a program system is a more complicated process which requires coordination of the efforts of a large collective of developers. Program systems combine packs and have a unified data base and task control and data base control language. Development of a program system places rigid requirements on standardization and unification of individual blocks of the system.

A fixed condition in development of software is their circulability or mobility [8], i.e., the programs should not depend on the hardware utilized. The experience of developing the Grafika system showed that FORTRAN-4 with minimum number of subroutine-functions and specific operators such as DATA must be used for these purposes as the basic programming language. The software should be constructed on the small-module principle, which permits rearrangement of its structure according to hardware of different configuration and of the internal storage capacity. Additional difficulties related to the characteristics of operating systems are encountered when using some types of domestic computers. The operating systems of these machines permit configuration of program segments only by interfering in the body of the programs and this taks place in a different manner on different machines. A number of translators have no specific COMMON blocks.

The capability of developing universal algorithms and programs should also be taken into account in development of software, which is sometimes disregarded even in those cases when the design algorithms are rather similar. A large number of positoning, layout and configuration programs is known for designing printed-circuit card templates, microassemblies and integrated circuits and for design of schematic and wiring diagrams. There is also a set of subsystems for modelling digital, analog-digital and analog devices. In such cases it is practically possible to utilize a unified software structure which limits the set of basic program modules that realize the main algorithms and a set of service modules that take into account the specifics of the problem being solved.

Thus, the main requirements on the basic software of SAPR can be formulated: a unified simple user language, an integrated data base, high circulation ability, universality, the possibility of free accumulation and unified connection of program modules.

Let us enumerate some characteristics of the Grafika system that we feel meet the given requirements.

Main Characteristics of the Grafika System

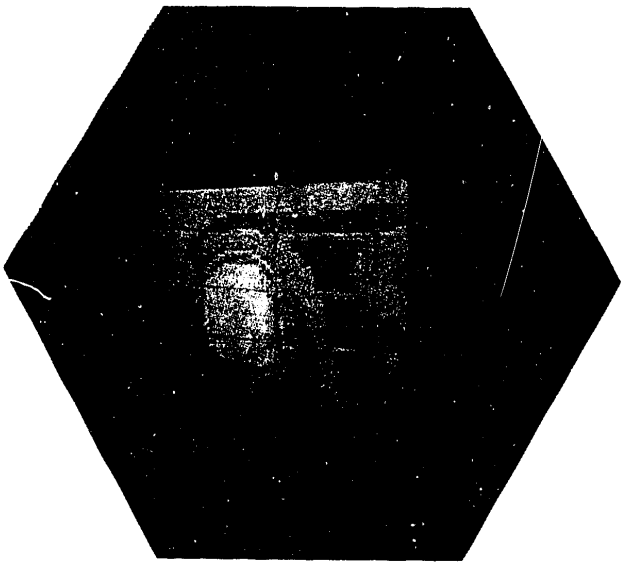
The Grafika system was developed on the basis of the ICL-4-70 and M-6000 computers and is now supplied for all types of YeS EVM [Unified computer system], M-4030, M-400, SM-2, Mitra 15/38 and Kongsberg machines. The system provides communications with various types of external devices; the SIGD vector display, three-color Tompson display, the TsGT color scanning terminal, EM-703, EM-712, KPA-1200, YeS-7054, Benson, Printronix, Kingmatic and Numericon graph plotters and coordinate graphs, EM-709 graphical information reader, EM-549 photo composing device and individual types of drilling and cutting machines.

The Grafika system permits solution of the problem of synthesis and analysis of the structures of specialized computer devices, specialized software, arrangement of components on a plane, layout of the connections between components, formation of a data base and input, conversion and output of drawings for a wide range of external devices.

The system can be used independently or in combination with applied software. To do this, applied programs may be connected to a special, so-called "dynamic" input of the system without interfering in the software or directly into previously stipulated locations. The system operates both in the pack and interactive modes. It is used to design specialized computer devices, to produce both diagram and design documentation for these devices, to design BIS and magnetic domain memory templates and to design and manufacture individual types of parts. The graphical part of the system is used in a geophysical complex (SM-2 and PS-2000) as a visualization system; examples of display of geophysical information on a TsGT screen and a matrix type printer are presented on the cover page.

Let us also consider some principles that are the basis of selecting the structure of the software of the Grafika system.

Synthesizing the Structure of SAPR Software



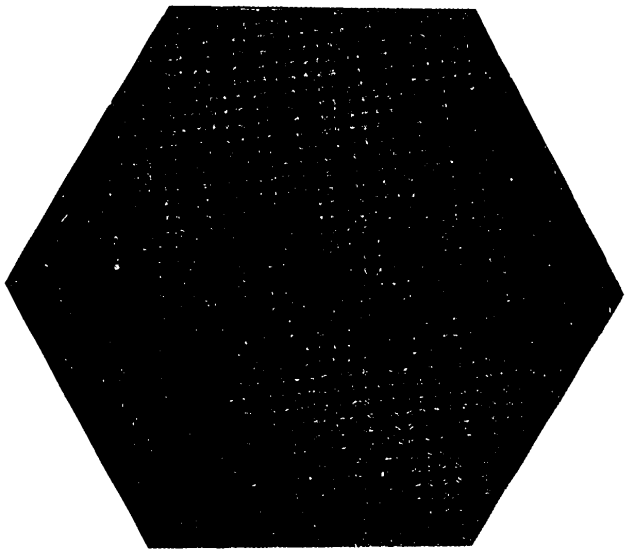
Example of Printing Geophysical Information on Color Graphical Terminal

Synthesizing the structure of SAPR software is an important component of the process of developing these systems which significantly affects the area of use, prospects and possibility of free accumulation.

Let us formulate the main task of synthesis on the basis of the following example. Let us assume that the software is designed to perform the following functions (Figure 1).

1. Entry of drawings to the computer memory (block P1), editing by means of functional keyboard FK, display on the graphical display screen GD with memory P1' and making the accompanying calculations V1.

2. Entry into memory P2 of the computer from the graphical information reader SG, making calculations V2, storage (block P2') and retrieval of information to coordinate graph KF, for example, to produce highly accurate documents.



Example of Printing Geophysical Information on Parallel Type Printer

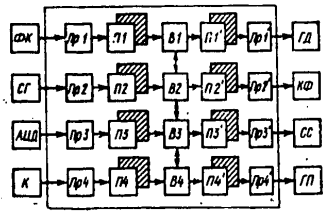


Figure 1

3. Entry into the memory P3 of the computer from the keyboard of the alphanumeric display ATsD for describing parts in terms of the input language, making calculations V3, recording the results in memory P3' and preparation of control information for drilling machine SS.

4. Entry into computer memory P4 from teletype keyboard K for describing the model of the device, modelling of calculations V4, storage of the results of modelling (block P4') and retrieval of the functioning diagram to graphic plotter GP.

Converters Pr_i are used at the input and output of the system to coordinate the internal and external information display. Each block of the internal storage P_i is accompanied by a nonvolatile memory (the cross-hatched rectangles in Figure 1) for storage of standard drawings, standard descriptions of models and so on.

It is required that the structure of the system optimum in memory, speed, number of program modules (or operators of the basic programming language) and the number of contacts between modules be determined.

In the general form, one of the versions of constructing the system may be represented as shown in Figure 1, i.e., in the form of a matrix of program blocks which perform the corresponding operations. Each i-th program block is characterized by accuracy of display δ_i, the method of encoding a_i, the form of information presentation ϕ_i and the functioning algorithm A_i.

Publications are now known in which systems that realize only individual lines of the indicated matrix are described. These solutions cannot be recognized as satisfactory since they make subsequent development of the system difficult. Combining the lines of the software matrix permits one to solve in a complex manner problems of design and provides access of each output device to each input device. This combination is possible due to the connections of blocks V_i and to some complication of them.

The structure shown in Figure 1 has considerable redundancy due to the large number of blocks V_i in which identical operations can be found and due to the considerable number of converters at the input and the increased number of programs which serve to organize the nonvolatile storage of information. The total number of blocks can be reduced by combining certain blocks. The following rules should be used when combining two blocks: the accuracy of information display in combined blocks is selected as the larger of δ_i, δ_j [$\delta_0 = \max(\delta_i, \delta_j)$]; the method of encoding and the form of information display in a combined block are determined by the external devices; and the resulting structure should not have a speed less than that previously given.

One of the versions of sequential combination of blocks is shown in Figure 2. In this case all the calculation blocks are combined into a single block V and natural display of information not dependent on the codes of the external devices is employed. The accuracy of display and the method of encoding are determined by the display for blocks connected to the graphical display.

The combined memory blocks at the input of the system Pvkh and selection of information display in terms of the user's input language permits one to significantly reduce the memory and conversion blocks. Only the converter PrSG remains, since in the general case the reader codes may not coincide with the user's input language. The memory block at the output Pvykh can be reduced if a unified method of encoding the output graphical information is adopted. It is obvious that the form of information display related to modelling and to the graph may differ; therefore, an independent block for storage of the results of modelling PM remains. Joining of

blocks Pl and Pvykh is known. This leads to an increase of the accuracy of display of the combined block, which will be redundant for the display. However, this version makes the system oriented to specific hardware.

Solutions are known when some functions of block V are transferred to the conversion block at the input (output) of the display PrD. In this case a group of blocks is formed determined by the dashined line in Figure 2, which can be used independently for input, output and performing individual operations on drawings. This group of blocks is called an intellectual terminal.

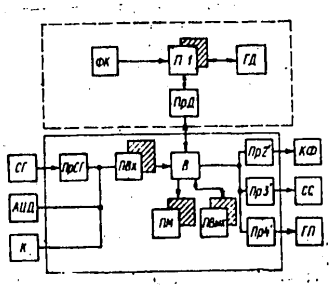


Figure 2

Combining the output conversion blocks is actually impossible since parameters δ_i, a_i and ϕ_i of these blocks differ considerably. However, analysis of the internal structure and algorithms of the conversion blocks led to an interesting trend in programming--design of adjustable information integration blocks [9]. Adjustable integration information blocks significantly reduce the time to connect new external devices to the system being operated.

Consideration of the example was begun with preliminary separation of the algorithm of functioning of the entire system into large operations. Let us attempt to determine this process somewhat more precisely. Let us introduce the concept of "local" algorithm. Let us call the connecting algorithm of sequential performance of mathematical operations characterized by unified accuracy, form of information display and one that utilizes a single method of encoding a local algorithm. Then the initial separation of the algorithm of functioning of the entire system can be carried out if $\delta_i \neq \delta_j$, $a_i \neq a_j$, $\overline{\varphi_i} \neq \varphi_j$, $A_i \cap A_j \neq \varnothing$.

It should be noted in conclusion that the discussions presented above were taker into account when designing the structure of the Grafika system. A more detailed description of the structure and description of the user language is presented in [10].

Conclusions

1. Automated design systems are a complex of hardware and software. Development of SAPR requires coordination of the efforts of a large number of engineers, programmers and systems engineers.

2. Development of basic software for interactive wide-use design systems is now possible.

3. Implementation of measures to standardize and unify individual program modules is required of SAPR developers.

4. Programming should be carried out in minimum FORTRAN-4 language to develop software with circulation ability (mobile software).

5. An important task is development of adjustable information integration blocks.

BIBLIOGRAPHY

1. Brown, N. L., ENGINEERING DESIGN GRAPHICS JOURNAL, Vol 42, No 1, 1978.

2. Garret, K., ENGINEERING MATERIAL AND DESIGN, Vol 22, No 6, 1978.

3. Bobkov, V. A., Ye. A. Golenkov and V. A. Perchuk, "A Graph Packet for the BESM-6 and Elektronika-100I Minicomputer Complex with UGD-43 Graphical Display," AVTOMETRIYA, No 5, 1978.

4. Vinokurov, D. I., "Sistema programmirovaniya graficheskogo dialoga DIFOR" [DIFOR Graphical Dialogue Programming System], Minsk, ITK AN BSSR, 1978.

5. House, C. H., ELECTRONICS, Vol 53, No 5, Feb 28, 1980.

6. Glazkov, I. M., "Oborudovaniye dlya proizvodstva fotoshablonov i vypolneniya operatsiy fotolitografii" [Equipment for Producing Photo Templates and for Performing Operations of Photolithography], Moscow, Sovetskoye Radio, 1975.

7. Bayakovskiy, Yu. M., Yu. M. Lazutin, T. N. Mikhaylova and S. T. Mishakova, "Grafor: A Complex of Graphical Programs in FORTRAN," Preprint No 90, ITM i VT AN SSSR, No 5, 1975.

8. "Mobil'nost' programmnogo obespecheniya" [Software Mobility], edited by P. Brown, Moscow, Mir, 1980.

9. Tremba, V. Yu., "Proyektirovaniye sredstv informatsionnogo sopryazheniya v SAPR. Avtomatizatsiya protsessov inzhenerno-psikhologicheskogo issledovaniya i proyektirovaniya cheloveko-mashinnykh sistem" [Design of Information Integration Devices in ASPR. Automation of Engineering and Psychological Investigation and Design of Man-Machine Systems], edited by D. I. Ageykin, Scientific Council on the Complex Problem "Cybernetics," Moscow, AN SSSR, 1980.

10. Artamonov, Ye. I., "Principles of Constructing Interactive Design Systems," Loc. cit.

6521
CSO: 1863

PROGRAMMING FUNDAMENTALS FOR THE UNIFIED COMPUTER SYSTEM

Moscow OSNOVY PROGRAMMIROVANIYA DLYA YEDINOY SISTEMY EVM in Russian 1980 (signed to press 23 Sep 80) pp 2-4, 332-336

[Annotation, foreword and table of contents from book "Programming Fundamentals for the Unified Computer System", by Viktor Davidovich Aynberg and Yuriy Veniaminovich Geronimus, Izdatel'stvo "Mashinostroyeniye", 50,000 copies, 336 pages]

[Text] Computers of the Unified System (YeS) replace second generation computers. The YeS EVM [Unified Computer System] is a family of computers with compatible programs intended for scientific-technical, economic, administrative, and other special tasks. They are used in various automated control systems and in data processing systems. This book on programming is based on the technical and logical possibilities of the YeS EVM and its software. It provides basic information on the logical structure and operating principles of the YeS EVM, it presents the basic resources of the Assembler language, it examines FORTRAN-IV programming in detail, and it describes the DOS YeS operating system.

The material's presentation is accompanied by examples of programs written in Assembler and FORTRAN, and by examples of jobs performed with such programs using the resources of the DOS YeS operating system.

The book is intended for engineers and operators using the YeS EVM system, and primarily for beginning specialists; the book would also be comprehensible to persons with a secondary education.

Contents Page

Part 2
Logical Organization of YeS Computer Models

Part 3
The Fundamentals of Programming in Assembler Language

Foreword

The YeS EVM is a family of program-compatible third generation computers intended for a broad range of scientific-technical, economic, administrative, and various sorts of special tasks.

Industrial assimilation of the YeS EVM has been the result of the joint efforts of socialist countries represented in CEMA--Bulgaria, GDR, Poland, Hungary, USSR, and Czechoslovakia.

The YeS EVM includes several models differing in productivity, from the YeS-1020 (with a speed of 20,000 operations per second) to the YeS-1060 (with a speed on the order of a million operations per second).

A unified software system was created for YeS computer models with an eye on program compatibility, and it is continuing to develop. This software system controls all of the system's hardware, and it contains the resources for automation of programming operations and accumulation of libraries of programs, both ones intended for general use and problem-oriented applications programs.

The literature on the YeS EVM is oriented for the most part toward the reader with experience in programming other computers.

The experience of the authors in education has shown that a certain subset of the logical resources of the YeS EVM can be fully applied to the training of beginners.

This book immediately introduces a reader totally unfamiliar with the subject to the mainstream of the technical and logical possibilities of the YeS EVM, to include its software.

Much attention is devoted in the book to the basic programming concepts. The material is accompanied by a large number of examples.

The authors hope that the book will also assist specialists who already have programming experience with other computers to learn how to work with the YeS EVM.

The book examines not the complete system of instructions of the YeS EVM, but only a certain part, enough to write more or less substantial programs. Only basic information is communicated on the architecture and logical organization of the YeS EVM, and on the concepts associated with program execution and channel operation. In precisely the same way, the concepts and resources of Assembler language are not presented in their entirety. This approach makes it possible, in the opinion of the authors, to avoid cumbersome details that could overshadow the basic ideas. The authors also believe that refraining from a complete description of the logic of the YeS, its system of instructions, and all resources of the Assembler language is justified because the general user will not as a rule use the computer-oriented language for his tasks, turning instead to languages of a higher level. The novice reader hoping to become a systems programmer will be able to use the information on instructions and on the Assembler language in this book as a basis for studying more-specialized literature, for example (1, 16).

FORTRAN is one of the high-level languages intended for applications programs written to solve computation problems. Programming with FORTRAN algorithmic language is presented in the book in detail. A reader who assimilates this presentation will be able to write all programs he might need for his professional uses in FORTRAN. Concurrently the reader will be prepared to understand the more-complex programming languages of the YeS EVM, primarily PL/1.

Knowledge of a programming language alone is not enough to permit access to the computer. The programmer must understand the resources of the operating system with which he debugs and runs the programs he writes, and with the assistance of which he runs prewritten programs. The book presents the basic concepts of the DOS YeS operating system, which has enjoyed broader use with more-recent YeS EVM models than the more-developed OS YeS operating system.

The information on the DOS presented in the book and the information on FORTRAN are enough to support qualified practical work.

Concurrently, an acquaintance with the possibilities of the DOS will make it easier for a reader desiring to broaden his knowledge to study the resources of the OS YeS.

Parts 1 and 3 were written by Yu. V. Geronimus, parts 4, 5, and 6 were written by V. D. Aynberg, and Part 2 was written jointly.

[139-11004]

11004
CSO: 1863

SOFTWARE FOR M-400 MINICOMPUTER OF VTsKP OF UZBEK ACADEMY OF SCIENCES

Tashkent VOPROSY KIBERNETIKI in Russian No 108, 1980 pp 120-126

[Excerpts from article by I. Pulatov, B. M. Mukhamedov and M. U. Rakhimova]

[Text] One of the main objectives of the development of the time sharing computer center (VTsKP) of the UzSSR Academy of Sciences is to provide an opportunity for clients to edit and debug programs without the use of intermediate information carriers (punch cards and perforated tapes). The client is equipped with a "Videoton-340" alphanumeric display (ATsD), from which programs and raw data can be fed into a BESM-6. This enables the client to debug a program on-line and quickly. Connection of the "Videoton-340" display directly to BESM-6 is not effective, since it takes 15-20% of the BESM-6 processor time just to feed in and print out information from display terminals. Therefore VTsKP of the UzSSR Academy of Sciences has a bilevel hierarchical structure (Figure 1), on the top level of which is a large computer of the BESM-6 type, and on the bottom level a minicomputer of the M-400 type.

The M-400 minicomputer is a data concentrator and it performs the functions of a processor to which the subscribers are connected. In addition, some of the functions, related to the conversion of information, are transferred from BESM-6 to M-400.

An interrupt signal comes from the terminals every 800 μs. All of the programs of the operations system are written in ASSEMBLER M-400 language.

It may be said on the basis of the material presented above that by connecting terminals through minicomputers it is possible to make more efficient utilization of the useful time of a large computer; the operations system is so developed that it is easily supplemented with new programs without major changes; the bilevel structure of VTsKP gives other machines and terminals access to BESM-6 through M-400 without major expenses and without altering the BESM-6 operations system.

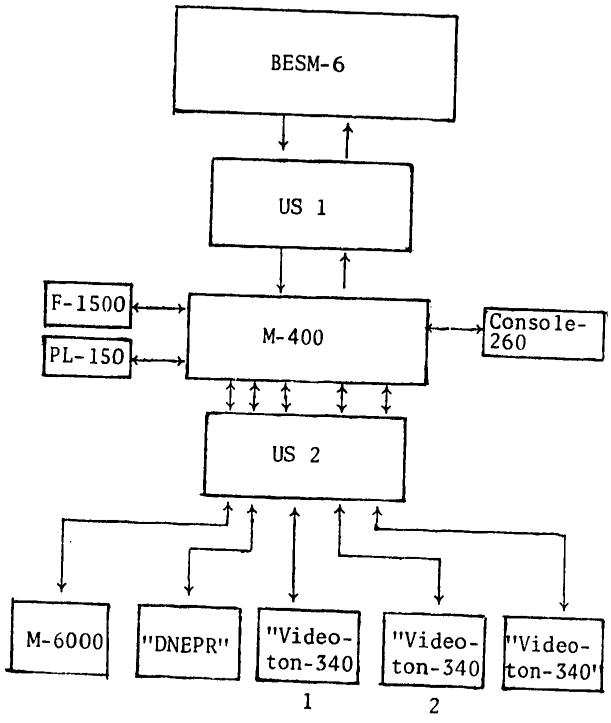


Figure 1.
[US -- Interface]

[129-7872]

7872
CSO: 1863

UDC 681.3.06:658.012.011.56

PRINCIPAL DIRECTIONS OF DEVELOPMENT AND STANDARDIZATION OF LANGUAGES AND PACKAGES OF APPLIED PROGRAMS FOR AUTOMATED PROCESS CONTROL SYSTEMS

Moscow IZMERENIYA, KONTROL', AVTOMATIZATSIYA in Russsian No 1, 1981 pp 57-65

[Article by V. K. Sheremet'yev, engineer]

[Excerpts] Development of Languages and Packages of Applied Programs for Automated Process Control Systems in the USSR in the 1970's

The principal stages in the development of languages and packages for automated process control systems in the USSR. Two periods can be distinguished in the development of languages and packages in the 1970's:

1) 1970-1975--search for the directions of development and the creation of the first Soviet systems permitting software mass production;
2) 1975-1980--systematic work on software standardization, including the creation of libraries of standard algorithmic and program modules, the development of automated software generation systems, etc.

By the start of the 1970's a number of original automated process control systems had been developed in power engineering, chemistry and metallurgy. As a rule the programming has been done in machine codes and in Assembler; high-level languages were practically not used, in particular, due to serious limitations of the machine memory volume and an absence of corresponding translators. The operating systems had limited possibilities and at times were developed by users for specific systems. The software was not designed for mass production even for systems related in technology and therefore, when the problem of mass production of leading models of automated process control systems arose, the cost of the programming work rose sharply. More and more new organizations, at times without adequate experience, have been drawn into work on automated process control system programming, and the general level of that work, in contrast with the programming of scientific and technical problems, was very low.

At the start of the 1970's at the Severodonetsk Scientific Research Institute of Control Computers (NIIUVM), the State All-Union Central Order of the Red Banner of Labor Scientific Research Institute of Complex Automation (TsNIIKA), and later in the Experimental Design Bureau of Automation (OKBA) and a number of other organizations they started work on the automation of automated process control system programming. By that time the main direction in the development of work on automation of software development was the creation of problem-oriented languages and

packages. In the Severodonetsk NIIUVM a system was developed with the input language OSKURT (Opisaniye Sistem Kontrolya i Upravleniya Tekhnologicheskimi protsessami--Description of Systems for Monitoring and Control of Technological Processes). This system consists of a package of applied programs with an input language of the "fill in the form" type [33]. The starting data needed for processing the parameters is recorded on forms of two types. The main form contains information needed for the interrogation and required primary processing of the technological parameters. Information on types of processing not required for all parameters is contained on a supplementary form: checking for technological reliability, smoothing, correction of expenditures and realization of the principal laws of direct digital control.

In the TsNIIKA work was started on the creation of standard automated process control systems. A component part of that work is automation of the planning of functional, algorithmic and program software for automated process control systems. In particular, a language was developed for description of algorithms for monitoring and calculation of VPVI (Vvod, Pervichnaya obrabotka, Vyvod, Informatsiya--input, primary processing, output, information) indicators, one serving as the input language of a problem-oriented programming system. The language resources include five forms containing starting data for organization of the input, processing and output of information to displays. The processing form has some means of nonstandard procedural processing which use apparatus of macrodefinitions [34].

The first systems constructed with use of a data base [35,36] have appeared. The basic idea of data base organization consists in the separation of means of linguistic declaration, the conversion and arrangement in the memory of programs and data processed by those programs. This permits assuring independence of the changes made in programs and data files. The possibility has arisen of expanding and changing the set of tasks accomplished by the system in proportion to its development, which is especially important for automated process control systems.

Programs cease to be dependent on data of a specific object, which permits creating systems tuned to the data of various objects, and by ths same token solving the problem of mass production in automatic process control systems.

By the mid-1970's a certain amount of experience had been accumulated in the development of mass-produced software systems and modern hardware and operating systems of machines with program continuity (the YeS EVM, ASVT and SM EVM) permitted accumulating and improving packages of programs in the course of many years.

Work on software standardization has become systematic. In the TsNIIKA the method of mass production of automatic process control systems--the creation of systems of standard solutions--has been developed. Systems of standard solutions have arisen as a form of generalization, unification and distribution of experience in automatic process control system development. Standard solutions are being developed in the area of hardware and of organizational, informational and mathematical software. In the area of software standardization, work is being done on the creation of libraries of program modules, input languages of packages of applied programs, data base structures and methodical materials on system generation and arrangement.

Work on software standardization includes automated process control system classification, the creation of standards, the establishment of a single terminology, the compilation of catalogs and descriptions of algorithmic and program modules and the creation of systems for packages of applied program generation.

Intrasector work directed toward the coordination of software developments for automated process control systems of various ministries have also been developed. The USSR State Committee for Science and Technology has adopted special resolutions envisaging, in particular, the development and introduction through the State Fund of Algorithms and Programs of standard decisions on automated process control system software.

Coordination work plans have been created and the leading organization and user organizations have been designated for each sector. The scientific organizational leadership of the work is performed by the TsNIIKA, and responsibility for the running of the program as a whole has been entrusted to the Ministry of Instrument Making, Automation Equipment and Control Systems.

The main content of the work is the compilation of collections of algorithmic modules by sectors, the creation of libraries of program modules and proofreading equipment of libraries and the development of problem-oriented packages of applied programs for computer-assisted generation of automated process control system software [37].

Main directions of the development of problem-oriented packages of applied programs. In proportion to the development of problem-oriented packages of applied programs the following main directions have been formed:

--systems with program generators;
--mixed systems with program and data base generators;
--systems with data base generators.

The transition from program generators to data base generators involves increase of problem orientation, a higher degree of adjustability and complication of the data structure. Systems constructed by means of program generators have a number of merits and shortcomings which make them similar to systems constructed on the basis of procedural languages. A problem orientation of systems is clearly reflected in data base generators. On the methodological level the transition from "pure" program generators to data base generators or mixed modifications of them involves a transition from the modular construction of systems "upward" toward a structural method based on the construction of program systems "downward." The modular approach was developed in the first stages of systematic work on software standardization, which involved the creation of collections of algorithmic and program modules. The creation of program module libraries requires, in turn, the development of program generators uniting those modules in systems.

Program generators assure the ordering of information communications between modules and data to be processed, and also the formation of leading programs which define the starting order of individual modules and the linkage of leading programs to the operating system. A traditional approach based on program compilation is being developed in systems with program generators. Two stages of compilation are used in those systems. In the first stage the program, written in a problem-oriented

language of modular description, is translated into a standard process control computer language (FORTRAN or Assembler). In the second stage ordinary compilation is done on the level of the machine language.

Difficulties of the modular approach are connected with the fact that individual modules are being developed by separate organizations, starting from different traditions and explicit or implicit assumptions regarding the structure of data to be processed. Program generators in the main produce ordering of fairly systemless informational connections, which leads to inefficient use of machines (of the memory as a whole, memory assigned to the zero page, machine time, etc) during work on a real-time scale. Attempts to impose more important limitations on the informational and control connections of modules lead gradually to the idea of creating program interfaces which when treated carefully exert an ever-greater influence on the structure of the modules themselves and require radical reprocessing of them.

The development of the systems approach to the creation of program interfaces is becoming a key problem of standardization. If an analogy is drawn with the development of modular technical structures (ASVT and SM), the main problem in them is not the structure of the separate devices but the structure of the standardized interfaces uniting those devices in a system. Standardized communication channels (the common bus, the 2K communication channel in the ASVT, CAMAC, the Hewlett-Packard instrument interface, etc) permit independently developing separate devices and connecting them to standrad interfaces through which control and informational connections are made. Then the data base can serve as a global interface in software.

Program generators. The GPO-A programming system is a typical program generator. This system was developed for the M-4030 universal computer and consists of a macrogenerator with the basic FORTRAN language of the M-6000 [38].

The basis of the work of the system is the principle of arrangement of the tasks of specific ASU's from previously programmed modules. Representation on the blanks on block diagrams of automated process control system tasks in the language of modular description is the generation task. Resulting from generation are the formed tasks, which have a controlling interface with an operating system. The tasks represent a sequence of requests in FORTRAN of program modules. The informational connections between modules are formed in the common region of the memory during generation.

The system described in [39] also is a program generator. A special mechanism for processing events which assures logical control of tasks is a distinctive feature of it. The linguistic means of generation are divided into the language of task composition and the language of subsystem organization. The language of task composition is a means of describing block diagrams and is used for the formulation of supervisory tasks which assure the request of individual modules and their informational connections. The language of subsystem organization assures creation of a mechanism of control within the framework of the subsystem, depending on the onset of certain events.

Tasks are composed from modules on the basis of the description of the task in the composition language. The generated task consists of a series of requests of individual modules. During composition each module is tuned to the field of parameters of the task organized in the process of composition, and the external

informational connections of tasks are also formed through the general region of the memory.

Mixed generation systems. In mixed systems the process of separation of programs from data starts, and this permits controlling their connections on a real-time scale.

Mixed generator systems. In mixed systems the process of separation of programs from data starts, and this permits controlling their connections on a real-time scale. This gives systems an interpretive character. Data structures are being complicated. In contrast with programs accomplishing scientific and technical tasks, in information systems, planning automation systems and ASU the center of gravity shifts from the program structure to the data structure; this results from their large volume and variety of types. The quality of the data structures serving as the starting material for data base construction is determined to a considerable degree by the characteristics of the entire system. Three generations of data structures used in automated process control systems can be distinguished:

--linear single-level successive structures in which the data elements are arranged one after the other in the order of their request for processing;

--mixed two-level structures in which tables are added to sequentially organized data for the retrieval in them of elements belonging to a single object;

--list tree-like or tabular structures which permit constructing multilevel nonlinear structures and by the same token depicting in the data bases fairly complex informational models of objects [40].

In program generators, as a rule, linear successive structures are used. In mixed generation system the second generation of data structures is used. In that case the data base does not contain controlling information, or contains only statically controlling information introduced during generation of the system. By data base is sometimes understood the set of parameters of adjustment of program modules [37, 41].

An example of a system in which data base elements appear is the ASP-RV [not further identified]. The input language of that system is similar to the language of macroinstructions and includes an operators section (of computation formulas), a variables section and an informational and logical connections section. Block and graph diagrams of an algorithm for the functioning of automated process control systems serve as the starting information for linguistic description. The system is adjusted on two levels, the logical and the informational. As in the system described in [39], logical adjustment assures control of tasks in real time as a function of approaching external events. The generated software represents a combination of a program complex and adjustment tables. The adjustment tables are used by the task interpreter, who seeks in the data base an adjustment table corresponding to the task to be actuated and successively the functional modules indicated in it, communicating to it the parameters of adjustment according to the mode of execution and that data.

The Severodonetsk NIIUVM has developed a package of program modules of generation of tasks in data collection and processing in automated process control systems, a package which has preserved some continuity of data structure with the system

based on the OSKRUG language [43]. The package is a generator of tasks, each of which has its local data base. By task here is understood the leading program, which is actuated by the operating system. A macrolanguage is used as the input language. By means of it internal tables of the system are constructed, representing within the limits of each task hierarchically connected declarers of the group and individual characteristics of the technological parameters. Interpreters introduced into each individual task are used to actuate each processing module.

Set as the basis of package construction are the principles used in the construction of modular software systems [agregatnaya sistema programmogo obespecheniya--ASPO] for series SM computers. The package is accessible for wide use. A step backward in comparison with the OSKRUG are failure to realize functions of direct digital control and the transition toward a macrolanguage from a more convenient language of the "fill in the form" type. Use of a macrolanguage gives rise to no special difficulty to programmers of systems generating operating systems (all the more so because a small volume of starting data is used there), but leads to complications when a larger volume of starting data is introduced for the primary processing and presentation of information, since unqualified programmers or system users are usually engaged in that. The main shortcomings of the system are connected with the fact that the introduction of data base elements is not carried out systematically enough and is not accompanied by a convenient data base control system. In particular, data base control systems for automated process control systems must assure means of access to data bases on the physical and logical levels.

The use of physical access requires knowledge of data base structure (disposition in the memory, structural unit dimensions, etc) and assures a high rate of approach. During logical approach the programmer uses symbolic designations of data elements. Logical approach is less effective but more convenient, as it does not require knowledge of the data base structure. In the package under consideration, means of convenient logical approach are lacking, whereas it is precisely means of logical approach which characterize the durrent data base [44]. Programs in the package are not separated from data. Information is retrieved by name by systematic sorting of all declarers until the needed digit of the parameter is encountered. Lack of development of methods of approach leads to low reactivity of the system.

A distributed data base can be the basis for use in decentralized technical structures but, in view of closure of the data base to a narrow circle of tasks, it cannot serve as a general system program interface. Due to absence of a unified data base control system the package suffers from redundancy. Interpreters are duplicated in each task. The number of tasks can be very large, as they are connected with particular sets of parameters related in types of processing and time of interrogation.

Data base generators. Methods of using data bases in automated process control systems are formed to a considerable degree under the influence of the development of data bases in general-purpose information systems [45]. The specifics of data bases for automated process control systems consist in higher requirements for the rate of access to the data base core, resulting from functioning of the system on a real-time scale with a small time of allowable delays. In view of the relatively small volumes of information and high requirements for reliability of the system it is better to place the data base in the main storage, and not on disks.

The small volumes of main storages and the requirement of maximally efficient work of the system on a real-time scale lead to division of data base formation into two stages: the static, accomplished in an off-line mode, and the dynamic, for data base correction on a real-time scale. Static generation requiring greater computer resources can be accomplished on an object machine using disks or on a universal computer. After the introduction of a generated data base in the main memory of a minicomputer with small resources, it is dynamically reorganized on a limited scale because of a need to introduce changes on a real-time scale. During basic reorganization of the data base it is additionally generated on another machine or by means of disks in background conditions.

As a rule a considerable part of the functions performed by data base control systems, especially those involving information retrieval, is realized during generation. This saves much machine time during work on a real-time scale.

The MEDIUS system has a data base generator [46]. Data are separated from the program here and arranged on disks; the classical system of files control is used. The data are divided into three types: constant or provisionally constant information, process variables and results of computations. The data structure is mixed and consistent for the results of computations. On the other hand, the constant and variable data are localized for each measured point. A fill in form language serves as the input language for data base generation.

The concept of the data base is used in the software system of the automated process control system for large-capacity units, the "SPO-Polimir" software system [47], consisting of a set of applied program packages developed for a large-capacity polyethylene production unit and combined data bases as a program interface. The basic principles of system construction are: separation of programs and data, the use of physical and logical methods of access to a centralized data base, the application of simple linguistic means for the static and dynamic adjustment of the data base, the hierarchic character of package structures, the modular character of programs and data. The applied program packages used in the "SPO-Polimir" are divided into two groups:

--packages connecting data bases with the external environment (technological process, technologist-operator);

--packages using a data base as a dynamic information model of an object.

The architecture of the construction of packages of the first group, accomplishing collection, primary processing and presentation of information and also direct digital control, is subordinated to the hierarchic principle: on the first level the interpreter of the inquiry enters dynamic information in the data base; on the second level the data base interpreter by means of a third-level library performs functions of the package; on the third level, libraries of problem-oriented modules are arranged. Packages of that group, to achieve high efficiency, are written in Assembler, placed in the main memory and work in a privileged mode in relation to packages of the second group, having higher priorities on the whole and using means of physical access to the data base.

Packages of the second group, accomplishing computation of secondary indicators, preparation of recommendations and optimum control, are written in FORTRAN, partially arranged on disks and use means of logical access to the data base. Their architecture is determined by the specifics of the package tasks.

A system data base consists of a core, consisting of a set of technical documents of measured and computed parameters, and also a description of the data base core structure which assures access to the core. Each technical document contains all the information needed for processing the given parameter. A distinctive feature of a data base is its activity. Each technical document consists of a set of units, the sequence of which represents a program of processing in relation to which the module plays a passive role. Thus the data control the requests of modules, in contrast with the ordinary circuit in which the interpreter communicates to the modules the parameters of adjustment or that connection has already been made during compilation (in the case of program generators). Programs controlled by data are becoming an important means of data base organization.

In contrast with the concept which has formed regarding the ineffectiveness of use of methods of intepretation in comparison with compilation, data base interpretation leads to a considerable saving of machine time, as it applies to the work of only the controlling part. This assures the possibility of work upon request, which is impossible when program generators are used. Program generators lack convenient means of dynamic adjustment of work of the system which are available (as a result of separation of programs and data) in systems with a data base.

From the methodological point of view the planning of systems based on a data base starts with the development of the data structure and the entire data base, which determine the program interfaces, internal for individual packages and external for the coupling of packages into the system. Simultaneously the entire software system is broken down into subsystems within which the levels and individual modules are determined.

Areas of application. At present all three directions of language and package creation are being developed, each of which is finding its own sphere of application.

Program generators, the functions of which are similar to those of standard software (macrogenerators and compositors) can find wide application as the problem orientation is weakly expressed in them.

Mixed generation systems are finding application in automated process control systems with a continuously discrete character of production for problems of operatively discrete control. In those systems the problem orientation is more strongly expressed but the set of functions and correspondingly of modules is rather broad and unstable.

Data base generators are used where high functional efficiency is needed and the set of functions is definite and stable. In particular, the creation of software for automated process control systems for large-capacity units is such an area.

Also possible is the combined application of data base generators for a limited circle of well developed functions and program generators for a broad circle of newly appearing tasks, where the set of modules and the data structure are still inadequately stable.

At the present time the TsNIIKA and ORGKhIM (the State All-Union Trust of the USSR State Committee of the Chemical Industry) are developing a passive main-memory

resident data base control system to store various sets of data in general purpose automated process control systems [48]. It can be used with the GPO-A program generator.

[119-2174]

2174
CSO: 1863

APPLICATIONS

UDC 658.012.011.56"313"

PROSPECTS FOR DEVELOPMENT OF AUTOMATED CONTROL SYSTEMS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 11-13

[Article by Candidate of Economic Sciences B. V. Karpov, chief of All-Union Production Association Soyuzsistemprom, and Candidate of Technical Sciences A. Ye. Rozinkin, chief engineer of VPO Soyuzsistemprom]

[Excerpt] The most important scientific and technical factor and the most inseparable link in the total system of measures to increase control efficiency were the ASU [Automated control system] of industrial facilities, sectors of the national economy and facilities of the nonindustrial sphere.

The effectiveness of measures in the field of improvement of control, including automation of it, can be evaluated by the final results of functioning of a facility equipped with ASU. Analysis of the results of putting an ASU into operation for different facilities confirms the high effectiveness of capital expenditures for these purposes. For example, the Ministry of Instrument Making, Means of Automation and Control Systems, has allocated a visible place to development of ASU in the sector of instrument building to support the very intensive tasks of the 10th Five-Year Plan among the complex of organizational and technical measures and economic measures. ASU have now been developed and put into operation at enterprises which produce approximately 85 percent of the general sector volume of product sales, which contributed to successful fulfillment of the tasks of the 10th Five-Year Plan by the ministry. ASU for laborious production processes--monitoring-regulating, metrological checking, assembly work and so on--have been introduced.

During the 10th Five-Year Plan, industry organized the output of new models of third-generation computers of the YeS EVM [Unified computer system] and SM EVM [International small computer system] series, with full use of the capabilities of which the ASU was developed. Scientific research and experimental design organizations created the necessary scientific and technical conditions (standard design solutions, applied program packs, program tools for automation of ASU development, methodical and guide materials for ASU development) which serves as a basis for development of ASU of a high scientific and technical level. The stock of standard design solutions and applied program packs developed in organizations of the VPO Soyuzsistemprom made use of more than 2,000 organizations and enterprises of different ministries and agencies of the country during the five-year plan in development of ASU of various designations. The organizations of the association participated in development of ASU for 1,100 machine building, light industry, construction,

transport, health, municipal management and other enterprises and thus made a significant contribution to increasing the efficiency of sectors of the national economy.

6521
CSO: 1863

UDC 681.3.008

THE PRINCIPLES OF DESIGNING COLLECTIVE-USE FUNCTIONALLY DISTRIBUTED SYSTEMS

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 16-20

[Article by Candidate of Technical Sciences A. A. Vdovin, director of Moscow Scientific Research and Planning Institute of Program Control Systems]

[Text] The level of development of computer technology achieved during the past few years, including remote information gathering and distributing devices, expansion of functions of common computer software, modern methods and means of developing special (applied) software, all in combination creates the necessary conditions for development of qualitatively new software-hardware computer complexes. These complexes are capable of providing information gathering, processing and distribution in real time for a large number (tens and hundreds) of territorially distributed facilities simultaneously, of working with considerable volumes of information (hundreds of megabytes) and of performing a wide range of functions: formation and management of large data bases, data processing and distribution and a reference interrogation mode for the entire aggregate of data. The distinguishing feature of these systems is specifically the completeness of collective use of the main systems resources--hardware, software and information.

Organization of collective-use computer centers (VTs KP) creates a basis for effective use of primarily hardware resources. Collective use of software and information resources becomes not only a possible, but a determining quality of large data processing systems oriented toward the use of organizational and economic activity in a specific area (sphere).

Examples of these spheres of activity, within the framework of which atomation of data processing is a necessary condition for conversion to a qualitatively new, more effective level of organization and management, are problems of topical information processing in the interests of many sectors, for example, management of technical and economic information classifiers, processing of earth sounding data and finally a complex of measures related to preparation and execution of a large purposeful program such as the 22nd Olympic Games of 1980 in the USSR.

The VPO [All-Union Production Association] Soyuzsistemprom, Minpribor [Ministry of Instrument Making, Means of Automation and Control Systems], are now conducting investigations to develop automated data processing systems. With regard to the latter example, the ASU [Automated control system] Olipiada developed by organizations of Minpribor, having successfully solved the tasks of management and information support of the 22nd Olympic Games, became the first complex automated collective data processing system (ASKOD) in our country.

The experience of development and operation of the ASU Olimpiada complex makes it possible to formulate the principles of design and to determine the main methods of realizing the software-hardware computer complexes under consideration, oriented toward application in a specific sphere of activity.

The Functional State and Structure of the ASU Olimpiada Complex

The main functional parameters of the ASU Olimpiada complex are the following: more than 100 sources--information users--are located at 34 Olympic facilities of Moscow, Tallinn and Kiev, the information about the 10,000 participants and about the results of more than 5,000 stages of the Olympic competitions, operationally used and presented in three languages (Russian, English and French), comprises more than 400 Mbytes, more than 200 information processing, conversion and distribution algorithms for competitions in 203 disciplines of 21 types of sports of the Olympic program are used, the results of the competitions are processed and reports of the results are distributed throughout the Olympic facilities directly during competitions and finally, special hardware and software make it possible to obtain information upon request about the Olympic participants and about the results of the competitions through information centers equipped at Olympic facilities in Moscow, Tallinn and Kiev.

According to the given requirements, the ASU Olimpiada solved four complexes of problems: processing the registration data about the Olympic participants and formulation of basic data files, processing operational data about the participants and the results of the competitions and formulation of starting, intermediate and summary reports, processing requests and formulation of reference data on the participants and the results of competitions and processing data to support the activity of the apparatus of the Olimpiada-80 organizing committee. The first three complexes of problems were determined by the requirement of real-time information support of all facilities of the system.

The basis of the concept of the functional and technical structure of the ASU Olimpiada complex were the principles of regionalization of functions and decentralization of the computer resources of the system. The principle of regionalization of functions includes the fact that such subsets of functions F_i for which the corresponding facilities (sources--data users) comprise a sufficiently small subset (region) N_i of the entire set N of facilities of the system are allocated from the entire set F of functions of the system. Each regional subset of functions should be sufficiently complete from the viewpoint of supporting the facilities of the region with the necessary data provided there is a temporary failure to realize the remaining functions. In this case the system which automates the subset of functions F_i may be regarded as an independent local ASCOD for objects of region N_i.

According to the principle of regionalization, seven regional subsets of functions (the number of special types of sport) were allocated from the entire given set of functions of the ASU Olimpiada. Each subset is an adequate set of functions for the corresponding type of sport in case competitions are held only for this type. Thus, the software and hardware complex that provides automation of any subset of functions may be regarded under specific conditions as an independent data processing system for the corresponding type of sport.

Regionalization of the functions and facilities made it possible to compile the structure of the hardware of the ASU Olimpiada on the principle of decentralization (distribution) of computer resources. According to this, a regional information-computer center (RIVTs), which is brought as close as possible to the facilities of the corresponding region, ensures realization of one or several regional subsets of functions F_i. Localization of solution of operational problems of data processing on the basis of the RIVTs permits minimization of the load on the data transmission channels and makes it possible to solve these problems autonomously within a region and in this case the results are in no way dependent on breakdowns and failures occurring outside the region.

Five regional information-computer centers--one at Tallinn and four at Moscow--were created in the ASU Olimpiada. The Luzhniki RIVTs provided automation of data processing of competitions in track events, gymnastics and water polo, the TsSKA RIVTs provided processing of competitions in fencing and wrestling, the Tallinn RIVTs provided automation of processing of the competitions in sailing, the OMP RIVTs provided automation of processing of competitions in handball and the Prospekt Mira RIVTs provided automation of processing of basketball competitions.

A main information-computer center (GIVTs) has been created to realize the functions not contained in the regional subsets and which are commonly used for all or most of the facilities of the system. This computer complex provides for development and management of a general systems data base, generalization, data processing and distribution at the regional level and finally an information mode in the general systems data base.

Development of an ASKOD based on a two-level hardware complex (the lower level is several RIVTs and the upper level is a GIVTs) has a number of significant advantages compared to centralized design of KTS [Hardware complex] based on a single computer complex. With a two-level structure of the KTS, there are more opportunities to increase the functional and operating reliability and viability of the system, to reduce the number and total length of data transmission channels, to parallel software work and to reduce the total deadlines for development of the system.

Realization of these capabilities during design of the ASU Olimpiada complex was related to investigation and solution of such problems as determining the level of mutual redundancy of the functions of the RIVTs and GIVTs, organization of distributed data bases and provision of information compatibility and communications of the RIVTs and GIVTs, selection of an efficient structure of the remote data processing network, method of testing the system and adjusting its parameters and resources to specific conditions and development of special devices for monitoring and protection against breakdowns and restoration of functions.

The problem of constructing those methods of developing the system, primarily its hardware and software, which permit one to minimize the dependence of the deadlines of developing the system on the degree of its complexity is of no less significance.

Configuration of the Main Information-Computer Center of the ASU Olimpiada

The main information-computer center of the ASU Olimpiada was constructed on the basis of a medium-class computer, remote data processing equipment and general systems software which ensure real-time operation of more than 100 territorially

distributed users simultaneously. The GIVTs contains two central processors, one of which is a reserve. Each central processor has one Mbyte of internal storage, one byte of a multiplex channel and four block-multiplex channels. The virtual memory concept realized in the processor expands the volume of its logical memory to 16 Mbytes.

Two sets of external storage devices, each of which includes six disk memory devices with capacity of 100 Mbytes each, three magnetic tape stores and one control block for each type of memory, are used in the GIVTs configuration.

The use of two-channel switches in the external magnetic disk and magnetic tape storage control devices and also the switches to the two control devices in the external storage devices ensure multivariant redundancy of these devices and makes it possible to restore the efficiency of the system in case the working processor fails by connecting the reserve processor without rearranging the information carriers.

Flexible variation of configuration is provided by means of a switching device through which the interfaces of the input-output control devices are connected to the channels of both processors. Distribution of input-output devices through the byte and block-multiplex channels of the processors provides parallelling of input and output information flows, increasing the productivity of the system as a whole.

The remote data processing equipment includes two program-controlled communications processors, of which one is a reserve, 177 synchronous modems and 139 terminal stations. Each communication processor has 38 channels, of which two are allocated for reserve, while 36 are used to connect the terminal stations. The table for connecting 139 terminals to 36 channels of the communication processor is designed with regard to the functional designation and intensity of data exchange of each terminal and also so that the different terminals at a single facility are connected to different channels.

Duplication of the devices of the central part and the capability of combining the functions of each terminal provides the viability of the system in case any computer device or data transmission channel fails.

The Software of the ASU Olimpiada GIVTs

An OS/VS version 6.7 with virtual memory is used as the operating system (OS) of the GIVTs computer complex. The basic telecommunications method of access (VTAM) is used to control information exchange between terminals of the network and computer. Access of applied programs to the data base is mainly accomplished by using the method of VSAM access and the direct method BDAM is also used.

The random flow of requests (transactions) for data processing coming into the computer from the remote terminals is processed in real time in the operating mode. Ordinary operating systems, including the OS/VS, do not have convenient high-level facilities at their disposal that permit organization of this processing. Specialized remote processing program systems should be utilized for these purposes. Selecting the specific remote processing system is one of the main factors that determine the subsequent nature of the entire development. Program systems of the "Data transmission/Data base" (SPD/BD) type, for example the Kama system, have

become widespread for solving telecommunications problems of data processing between remote terminals and the computer and for organizing the access of remote users to the collective data base. The virtual version 1.4 CICS is used in the given case.

The SPD/BD system is a combination of control and service programs, tables and systems sets of data. When performing a transaction, each access to the SPD/BD causes transfer of control of one of the control programs. Analysis of systems utilizing SPD/BD showed that the volume of the applied code carried out in this case comprises an average of less than 10 percent of the volume of the code of the SPD/BD programs used in processing the transaction. Hence, it follows that the functional capabilities of the operating mode are determined to a large degree by the functional structure of the control programs of the SPD/BD.

Individual program modules of the SPD/BD were modified by including specially developed programs in them or by introducing changes to the initial texts of the control programs. These changes basically had the purpose of improving diagnosis and of providing more flexible response to different breakdown situations.

Monitoring and Control Software

Both the operating system and the SPD/BD offer specific devices to users to operate large remote data processing systems. Nevertheless, the specific experience of development and operation of the ASU Olimpiada showed the inadequacy of standard systems devices for effective control of a large collective-use system. The need for more developed facilities for centralized control of the system with a large number of terminals in the operating mode, for operational restoration of the system after breakdowns, for control of the operation of long dialogue transactions and so on became obvious during the course of development. Development of a special supporting subsystem, which is the superstructure of the SPD/BD, was required in this regard.

The main functions of the supporting subsystem, which are of greatest interest to developers of these systems, can be divided into two parts: support of applied developments and support of the functioning of the system as a whole. Support of applied developments includes creation and management of special files, formulation and printing of output documents and servicing of applied programs. Support of the functioning of the system includes control of the terminal network, control of the system and restoration of the system components and of the system as a whole. These functions were realized by modification of the input module of the SPD/BD and by development of new modules which supplement and expand the capabilities of the SPD/BD and also those which offer new capabilities for development of applied systems.

The data base includes two general-purpose files essentially independent of specific applications: requests and commentaries. Because of its universality (the single restriction placed on the entered information is that this is a dimension of unitary reference), the reference file can be used in any system as an additional reference device also used for purposes of controlling the operation of the system. The commentary file consists of recordings of variable length containing two alternatively accessible texts in different languages. The facilities of the supporting subsystem permit entry of new recordings, modification or removal of old ones and interrogation of any recordings directly from the display screen in the on-line mode.

The recording of the commentary file also contains the keys of other recordings and the recording is expanded automatically by replacing the keys of other recordings contained in it by their text upon selection of this recording to the applied program (for printout or to the display screen). Thus, formulation of identical machine documents in different languages is provided. Moreover, there is the capability of automatically calculating the recording utilization frequency and the most frequently used recordings are placed in a special rapid-access resident buffer.

A special function of the supporting subsystem controls the formatting and printing of output documents. The advantages of using it include the simplicity of programming, the flexibility of document formatting, more rapid functioning of the printer itself and of the system as a whole, which is especially important in parallel distribution of information to many output terminals. The supporting subsystem has a large group of supplementary programs which perform a number of standard functions of applied system support:

internal sorting in which the sorting key and types of sorting are controlled parametrically;

line checking for alphabet uniformity which permits determination of a line either in Latin alone or in Cyrillic;

checking for the presence of impermissible symbols in the line upon transmission to devices having a limited set of symbols;

editing the lines according to the given format;

recording problems in a special log and sampling data according to different parameters: time, number of the problem or file;

debugging the applied programs in the on-line mode from a remote terminal;

queueing by organization of lines of terminals for resource (file) anticipation rather than lines of problems as is done in SPD/BD.

There must be special devices for control of the terminal network in systems operating with a large number of remote terminals. If the SPD/BD provides the capability of interrogating the state of one terminal or of a line, it is very difficult to gain a complete idea about a network of 150-200 terminals using standard SPD/BD devices. One of the functions of the supporting subsystem makes it possible to obtain a complete pattern of the state of the terminal network. Information about 200 terminals of the network is displayed in compact form on a single screen.

Another facility of the supporting subsystem--obtaining a page-by-page list of the terminals subject to breakdown, arranged by the decrease of the percentage of data transmission errors, on the screen--is used to diagnose the physical errors of data transmission and for preventive maintenance of terminals.

A special terminal for servicing the network is usually allocated in large remote processing systems. The service modifications made in SPD/BD modules provide for recording all the changes and messages concerning the status of the network on the

printer of this terminal, i.e., an operating log of the terminal network is essentially printed. The set of additional equipment for controlling the remote processessing system provides recording of all changes in the system made by the operator of the main terminal of the system: parameters, file status, programs and so on. Moreover, all connections and disconnections of the system operators and all emergency completions of problems in the system are recorded. Thus, the operator of the main terminal of the system (like the operator of the service network terminal) is always aware of all events occurring in the system. The function of gathering the statistics of the supporting subsystem issues a number of supplementary data upon comparision with a similar function of the SPD/BD: about the components of the system (programs, lines, terminals and so on) outside servicing, about the utilization frequency of resident modules, about some errors in the operation of the system, about the utilization frequency of commentaries, about the statistics of operator errors and other information.

There is a possibility of controlling the output of information by operational variation of the point of designation, redetermining the different physical terminals assigned to each point of designation, i.e., by modifying the table of designations of the SPD/BD in the on-line mode.

A special function of the subsystem provides for gathering data about the errors of each operator working with the system. Incorrectly selected parameters, incorrect use of functional keyboards and erasing the screen are taken into account. This device is useful in training terminal operators to check the course of training and to evaluate the occupational readiness of operators.

A significant part of the supporting subsystem is oriented toward restoration of the system or of its components in case of abnormal and emergency situations, which is a development or supplementation of the SPD/BD devices and of the operating system. A special error processing program analyzes the breakdown situation and seeks a detour around it in the case of program breakdowns. If the operation of the program cannot be restored, the corresponding applied program is removed. If a terminal fails, the system attempts to restore it to a working state, because of the failure and the function of the terminal in the system is analyzed and the message is transferred to the network servicing terminal if necessary.

The problem of preservation and restoration of the state of the system becomes timely in a developed remote processing system. The subsystem provides for storage of the terminal control table in a special file and restoration of the current state of this table, i.e., of the terminal network if necessary. The table of designations is restored in similar fashion. All these are extremely important with increased requirements on the operational nature and reliability of the system.

One of the most dangerous situations in the system is file destruction. The subsystem provides for procedures which increase the reliability of preservation of files and means to restore them.

Because of its independence of specific applications, the supporting subsystem can be connected essentially without changes both to new developments (completely) and to already existing systems (by individual functions). It use permits a reduction of the periods of development of systems, an increase of the reliability of their functioning and control.

Special Software of the ASU Olimpiada GIVTs

The experience of developing the applied software (ISMO) confirmed the effectiveness of the method used to develop applied programs, the structure of their design and the main principles of organization of applied software.

The method of programming utilized in the ISMO corresponds to such important conditions of developing large program systems (hundreds of thousands of instructions) as the modularity of programs, the illustrativeness of describing the program logic and convenience of documentation, simplicity of modification and expansion of programs, the high level of checkability of the system, the possibility of combining different stages of development, debugging and testing of programs and as a result the reviewability of the periods of developing the system as a whole.

The main components of this method are structural programming, development of programs "from top to bottom," development and use of a development support library, the group work of programmers and also such production procedures as use of program pseudo-codes, the use of hypograms, structural design and structural passes and inspections.

Special requirements are placed on applied programs designed to function in the operating mode, i.e., under the control of the SPD/BD. Let us determine some of the main principles of designing applied programs of the operating mode. All these programs should primarily be quasi-reenterable, which means the capability of simultaneous operation with a single copy of the program in the internal storage to problems of processing any amount, generally speaking, of different data. A significant reduction of consumption of computer resources can be achieved by arranging applied programs so that one program can participate in the work of an entire class of transactions. This is achieved by dividing the transactions code into several separate programs so that each of them is used during processing of other transactions.

A further development of these principles is development of resident applied programs similar to the control programs of SPD/BD. Resident applied programs frequently perform the functions requested in the system which touch on the most important resources and which require especially well-thought-out organization of work. Some of the functions are universal in nature; therefore, the programs developed for realization of them can be used as standard software designed for collective-use systems based on SPD/BD. These functions primarily include organization of information output to the printers of the network terminals. With a large volume of printing, attaching printout to resources problems for a prolonged period may cause catastrophic consequences.

The resident printout module in the ISMO provides for formulation, compression, storage and asynchronous printout of information upon requests from applied programs such that the problem checks only the resources required for this during printing time on the terminal. The next resident module provides automatic insertion of information instead of the key words and numbers of the athletes entered from the screen. It may be regarded as a universal means of working with a generalized glossary-codifier in the operating mode. The resident applied program for special file control provides upon request access from transactions to this file at the logic recording level. The fourth resident applied program for special control of

resources sets the terminal upon request from transactions in line for the resource stated in the request. It is used mainly to attach the desired resources to the terminal during pseudo-dialogue (the means of controlling SPD/BD resources are called problem rather than terminal resources).

Pseudo-dialogue--the most efficient method of operation from the viewpoint of system resource control which provides multiple data exchange between the terminal and processor within a single transaction--is used extensively in ISMO. It essentially consists of completing the task immediately after receiving a request to print out processed data to the terminal. Since correction or entry of new data from the terminal is characterized by a completely different time scale than subsequent processing of them by the processor, then freeing essentially all resources during this correction is significantly reflected in the operating characteristics of the system.

Organization of the Data Bases of the ASU Olimpiada GIVTs

Requirements of high operating quality and functional reliability of the ASU Olimpiada complex became the determining requirements in selecting the structure and methods of organizing the data base of the GIVTs. These requirements together with existing limitations of computer resources (one Mbyte of the main memory) and the nonuniformity of the structures of the main information facilities (more than 500 different types of protocols) made the use of universal SUBD [Data base control system] unfeasible. The problems related to structural nonuniformity were overcome by developing a unified language for describing the protocol structures, which made it possible to process all protocols by unified basic programs.

Achieving the required operativeness depends on the productivity of the software-hardware complex, which is determined mainly by the intensity of turning the pages of the virtual memory, the intensity of data exchange with the BD files and the SPD/BD working files and by the intensity of data transmission over the channels. A reduction in the intensity of turning the pages was achieved by standardization of the software when the same program modules are used in different transactions, by extensive introduction of pseudo-dialogue when the memory resource occupied by it is completely freed during the intervals between data processing in a single transaction and finally by purposeful adjustment of systems parameters.

The intensity of data exchange with the system files depends primarily on the structure and organization of the BD and therefore contains low reserves for increasing the productivity of the system.

The principle of multistage organization of data was employed to reduce the volume of data during exchange with BD files and to accelerate the exchange itself. Thus, for example, the registration data about the athletes and referees have three-stage organization. File FI 1 is formed in the on-line mode directly during entry of the registration data. A new file FI 2 with direct organization is created in the pack mode after restructuring, which becomes the main file of input data for further processing of the operating results of the competitions. Further, tables of the athletes (TIN_{nn}) by types of sport, the data from which are generated by OS and SPD/BD equipment, which is considerably faster than reading the file, are compiled from file FI 2 by a special pack task. File FI 1 in which the indices for file FI 2 are entered during restructuring, but from which the registration data are removed,

is used for access to file FI 2 during additional registration and in the reference mode. Multistage organization of files permits efficient application of the principle of information compression-expansion.

A significant fraction of information is entered in the system in the operating mode and is processed in compressed form, i.e., in the form of codes. These are the numbers of the participants generated by the system instead of their names, the codes of the types of sport and sports disciplines, codes of special commentaries and so on. All these data are recorded in encoded form in the main operating file of the protocols of results (FEV). These codes are expanded upon transfer of the protocols to the printout distribution files F 80 and F 84--two-stage organization of the results of protocol files. The codes are expanded by means of the files of TIN_{nn}, FTA and FSE tables and the commentary files with two-stage organization--CLB and FCO.

We note some additional principles of organizing the data base which made it possible to increase the productivity of the ASU Olimpiada GIVTs.

Those entries which can be modified simultaneously by several transactions from different terminals should be avoided in the operating modes. The line for modification of these recordings may significantly reduce the response of the system. Operating transactions should be made repetitive, avoiding matched modifications of many files by a single transaction. It is also important to note the effect of such factors of data base organization as the use of "rapid"-access files (direct and VSAM), physical distribution of different files on disk volumes and loading some frequently used recordings--tables--in the resident zone of the SPD/BD on the productivity of the system.

The problems of increasing the reliability of the data base of the ASU Olimpiada GIVTs was solved both by standard SPD/BD equipment and by software and organizational equipment specially developed for these purposes. The most effective of the standard SPD/BD equipment is the Bacout programs which permit elimination of the consequences of emergency completion of transactions and the "hot" restart, which eliminates the consequences of failure to complete transactions during emergency completion of the work of the entire system. Supplementary equipment includes duplication of all modifications of recordings on special magnetic tape (journalization of files), restoration of file copies by means of logs and regulated encoding of files to be modified. Special equipment is also used to protect data against operator errors during entry of information into the system.

Some Principles of Organization and the Method of Developing the ASU Olimpiada

Development of and putting the ASU Olimpiada complex into operation, which included a total of 14 computers of various classes and more than 200 terminal stations, for connection of which to the computers more than 600 allocated telephone channels and approximately 2.5 million instructions of the special (applied) software was used and for which more than 3,000 operating personnel were required, determined the need to develop special principles of organization and of the method of development. One of the fixed conditions was provision of checking the course of the work and management of it. The organizational principle of step by step development of the system, adopted in development of the ASU Olimpiada, completely satisfied these requirements.

According to this principle, the version of the system--from simple to more complex --are worked out sequentially after construction of the general concept. Each step is a complete production cycle of development, including testing, adjustment of parameters and conducting tests of the system. Four versions of the system were worked out in sequence during development of the ASU Olimpiada complex.

Stage expansion of functions from step to step, multiple adjustment of parameters and planning of resources facilitated the achievement of the required values of the main characteristics of the system, specifically those such as productivity and response time. An important circumstance is also the fact that step by step development of the system permits one to begin training of operating personnel under real operating conditions of the system at the early stages of its development.

A special method of descending testing, which played an important role in providing increased requirements on the functional reliability of the system at the 22nd Olympic games, was used during development of the ASU Olimpiada complex, along with improved programming technology.

The problem of testing the system, i.e., checking the completeness and correctness of realizing the given functions, was divided into three stages: checking the special (applied) software, checking the functional capabilities of the system and demonstration trials. Each stage of testing requires careful development of the scenarios of test situations of special software and also teaching and training of personnel who carry out work at all stages of testing independently of the developers of the system.

The use of methods of simulating computer complexes and data processing, development of an entire complex of modules and use of them at all stages of development ofthe ASU Olimpiada became an important factor in achieving a high level of productivity and reliability of the system.

Along with simulation models, full-scale simulation models which combine simulation of single components of the system with real operation of others, were useful and effective in investigation of the main functional indicators of the GIVTs. The effect of various changes in the configuration of the terminal network, in the files of the data base, in versions of applied programs and in systems parameters of OS and SPD/BD on the main indicators of the system was investigated as a result of a series of experiments based on a full-scale simulation model of the GIVTs. The use of a full-scale simulation model as a basis ensured eimplicity of experiments with high accuracy of the results.

The main functional indicators of the system under real conditions depend to a significant degree on the given values of the general systems parameters. These parameters in an operating system are lists of resident supervisor programs, the number of sections of virtual memory, configurations and number of external devices, the arrangement of systems sets of data on magnetic disks and the parameters of page exchange. These parameters in the SPD/BD are maximum number of simultaneously executed problems in the operating mode, the dimensions of the sections of virtual memory, the priorities of problems and terminals, the residency and dimensions of the working zones of applied programs, the dimensions and number of buffers for sets of data and the sequence of entries in control tables.

Modelling in combination with special measurement software made it possible to estimate the effect of different variants of values of systems parameters on the main functional indicators of the system and thus made it possible to adjust the parameters of the final version, i.e., to determine the best specific values of systems parameters with respect to a given criterion.

6521
CSO: 1863

UDC 681.3:181.4

APPLICATION OF MICROPROCESSORS IN INSTRUMENT BUILDING

Moscow PRIBORY I SISTEMY UPRAVLENIYA in Russian No 2, Feb 81 pp 48-50

[Article by Academician of Georgian SSR Academy of Sciences I. V. Prangishvili, deputy director of IPU (IAT) for Scientific Work]

[Text] Extensive experience was accomulated by the enterprises and organizations of Minpribor [Ministry of Instrument Making, Automation Equipment and Control Systems] during the 10th Five-Year Plan on extensive application of the microelectronics component base in instrument building articles. Due to the use of the latest advances of microelectronics and development of new progressive principles of obtaining, transmission, processing and display of information, good prerequisites were created for development of a new generation of modular complexes of GSP [State System of Industrial Instruments and Automation Equipment] of increased reliability and accuracy, low cost and with expanded functional and structural capabilities on the basis of microprocessor sets of BIS [Large integrated circuit].

Wide application of microprocessor sets of BIS, microcomputers and microcontrollers becomes one of the strategic trends for development of modern devices and computer equipment (SVT) for extensive and effective automation of processes in sectors of the national economy.

The use of microprocessor sets of BIS during the 11th Five-Year Plan is envisioned in all fields of development of instrument building equipment. The most extensive use of microprocessor equipment will be found in computer and peripheral equipment (in articles of the SM EVM [International small computer system] and ASVT-PS), in numerical program control devices (UChPU), in modular systems of information gathering, recording and processing equipment (ASPI), in modular systems of monitoring and control equipment (ASKR), in modular systems of electronic measuring equipment (ASET), in monitoring-diagnostic equipment, in telemechanics systems, in electronic time devices and sensor systems and in other articles whose technical and economic characteristics are being improved considerably by use of microprocessor equipment.

The prospects for conversion to microprocessor equipment in instrument building articles are determined by the significant reduction of the labor consumption of produced articles (by a factor of 5-10), by a reduction of their cost (by a factor of 2-6), by an increase of reliability (5-10-fold) and by a reduction of overall dimensions and consumed power (by a factor of 10-20).

Development of microelectronic functional devices, analog microprocessor BIS, analog-digital and digital-analog converters ensures development of modern devices for communicating with the facility, integration with general-purpose microprocessors for developing a number of electronic measuring devices, monitoring-measuring and diagnostic equipment and other devices with high technical and economic indicators. For example, development of specialized semiconductor analog-digital converters will make it possible to develop small competitive measuring devices without movable parts (instead of mechanical devices) with electro-optical scales.

The appearance and broad introduction of microprocessor equipment induces a significant shift in the strategy of control in the direction of decentralization and generates a new approach to construction of control and regulation systems and in this case the properties of decentralized and centralized control structures will be combined to the optimum extent. Distributed decentralized control systems having features of network organization and a high degree of integration in utilization of resources and which propose wide use of digital information processing methods at the points of occurrence with simultaneous decentralization of production process control become predominant. Unlike a centralized system, there is no clearly marked center of information processing in a distributed system and because of this there is a large number of active processing equipment capable of interfering in the operation of the system, which determines the new organization of control of the interaction of devices of the system.

Extensive use of inexpensive and reliable microprocessors and programmed microcontrollers in a distributed ASU TP [Automated production process control system] permits preservation of the main advantage of digital devices--free programmability and accuracy--and at the same time ensures the high viability typical for classical decentralized analog regulating systems.

The use of programmed microcontrollers which perform digital processing instead of classical analog regulators for automatic control systems permits one to eliminate the lag of practice observed for a long time behind control theory and to effectively realize any laws and algorithms of optimum control proposed by modern control theory and also stimulates development of more complex, but at the same time more optimum laws and algorithms of control. Each programmed microcontroller whose functioning is determined entirely by the program embedded in its memory, may replace up to 10-30 analog controllers. A microcontroller is controlled either automatically according to a given program or by the operator by using the operator-technician console in selecting the required algorithms and configuration of the control circuit, its dynamic adjustment, selection of settings, variation of the operating mode and so on [1, 2]. Moreover, the use of a programmed regulating microcontroller permits one to realize those processes which cannot be executed when using ordinary analog controllers, including a change in the laws of control during operation, preliminary processing of input signals, selection of optimum modes by calculation of parameters which are not subject to direct measurement, digital control with high accuracy, better stabilization of parameters due to the absence of zero drift and so on.

The use of adaptive microcontrollers is most feasible for industrial control systems that contain nonlinear components, components with time-variable parameters or with a previously unknown model and components with large time constants.

Standard controllers-regulators have principal deficinecies which do not permit their use for control of facilities having nonlinear parameters with variable amplification. Controllers-regulators with adaptive amplificaton, which automatically regulate amplification according to the real values of the parameters of the process constructed on the basis of digital electronic components and rigid logic are extremely complex and expensive and therefore are rarely used. Complex algorithms of adaptive control of amplification are realized by the program method, which is simpler and less expensive [3-5], in adaptive programmed microcontrollers constructed on a microprocessor base. Distributed control systems which use microprocessors and programmed microcontrollers provide a significant reduction of cost, an increase of reliability, viability and flexibility, productivity and speed of response to interrogation. Thus, the use of programmed microcontrollers instead of analog regulators makes it possible to reduce the cost of the control system by 15-20 percent in the power engineering, gas, petrochemical and chemical industries and by 10 percent in ferrous and nonferrous metallurgy. The use of programmed microcontrollers in ASU TP for local monitoring and regulation permits an increase in the output of annual products by 1-2 percent in chemistry and petrochemistry, an increase of the productivity of hot-rolling mills by 2 percent in metallurgy and conservation of fuel by 0.1-0.2 percent in power engineering.

Microprocessors and microcomputers will be used in measuring equipment as built-in computers and control blocks which enhance the accuracy and efficiency of measurements. Measuring devices based on microprocessors are sometimes called "intellectual" devices and are capable of automatically selecting the required operating mode, the optimum measurement error and speed and of accomplishing self-monitoring and self-calibration. The functional capabilities of the devices are expanded in this case, their maintenance is simplified and reliability is considerably increased. Significant growth of the nomenclature of measuring and recording devices with built-in microprocessors is planned during the 11th Five-Year Plan.

Microprocessors and microcomputers built into remote control equipment impart a new quality to remote control systems. Various methods of data compression to reduce the load of communications channels and the memory of the central computer, to accomplish preliminary processing of information, to perform additional functions of local automation devices and to accomplish adaptive search for efficient transmission modes and bypass routes upon failure of the main modes may be used due to microcomputers built into remote control devices. Forecasting of emergency modes, diagnosis of damage and debugging of automatic remote control systems are simplified.

Microprocessor devices find the broadest application in computer equipment in design of small and microcomputers and control computer complexes (UVK) in implementation of various general-purpose and specialized processors of intellectual peripheral devices, an expanded set of devices for communicating with the facility, information input and output subcomplexes and channels and so on.

Investigations to develop and assimilate a new class of UVK based on computers with uniform readjustable structure (PS-2000 and PS-3000 computers) were developed at organizations of Minpribor during the 10th Five-Year Plan. Geophysical expeditionary and regional computer complexes designed for rapid processing of the results of seismic prospecting for oil, gas and other minerals and other processing of data on study of the earth's crust and upper mantle to find deep mineral deposits and to predict earthquakes are being constructed on their basis. The named complexes

permit an increase of productivity in processing seismic prospecting information 100-fold or more and to achieve a large saving due to increasing the accuracy of deep drilling, discovery of oil and gas at great depths and optimization of deep test drilling.

Development of the indicated highly productive microprocessor computer complexes became possible exclusively because of extensive use of microprocessor equipment, microassemblies and modern microelectronic technology.

The low prices for microprocessors considerably alter the traditional method of organizing data processing systems. The most expensive components of the system are now the peripheral devices rather than the processors. Therefore, a need not inherent to traditional computer systems to share the time of the common peripheral equipment among several microprocessors has now appeared. Thus, unlike the ordinary time-sharing mode of processor operation in solution of individual problems, a mode when (expensive) peripheral equipment is utilized jointly by several (inexpensive) microprocessors is becoming reality.

The proposed mass use of inexpensive microprocessor equipment during the 11th Five-Year Plan will lead to revolutionary changes in the field of electronic instrument building and will apparently induce essentially new concepts in the field of information acquisition, perception and processing. The mass use of microprocessor equipment will become feasible in many fields of instrument building in which the use of computer equipment was previously regarded as unacceptable due to technical and economic concepts.

The use of microprocessor equipment opens up new capabilities in construction of different means of obtaining information (for example, sensors). Thus, a flow meter realized on the basis of microprocessor equipment permits effective solution of the problem of measuring large water flows with an error of no more than 1 percent compared to 2.5-4 percent of those now available.

The use of microprocessor sets of BIS, microcomputers and microcontrollers is especially important in those devices and systems which have a significant effect on increasing labor productivity and conservation of fuel and energy resources. Rapid growth of production of programmed microcontrollers and microprocessor control-computer systems and devices based on microprocessors for numerical program control of machine tools and program-logic control of various units, equipment lines and industrial robot manipulators, for regulation of the parameters of production processes and checking the measurement and diagnosis of product quality, for control of the search for production and consumption of fuel and energy resources and so on is expected in this regard during the 11th Five-Year Plan. Microprocessors and microprocessor systems will find the broadest application in machine and instrument building for program and logic control of metalworking machines, presses, units, equipment lines and industrial robot manipulators. The number of machine tools in the world equipped with numerical program (digital) control (ChPU) now comprises 3-4 percent. In this case ChPU devices based on microprocessors and microcomputers cost one-half or less than ChPU devices without the use of microprocessors.

Industrial robots or automatic manipulators with built-in microprocessors or microcomputers have achieved serious development. The stock of industrial robots for control of production equipment and production processes will increase most

intensively. Development of automatic manipulators or industrial robots with microprocessor control for complex mechanization and automation of machine and instrument building, the coal and coal-mining industry, ferrous and nonferrous metallurgy, agriculture, transport, construction and other sectors at accelerated rates in 1981-1985 is a problem of primary importance, solution of which will provide an increase of labor productivity in these sectors of the national economy.

Microprocessors and microcomputers built into industrial robot manipulators make it possible to utilize new methods of teaching by means of manual control. In this case the entire program of action of the robot is not previously recorded in the memory of the microcomputer, but the sequence of operations to be executed is entered operationally step by step. The instructions recorded in the ZU [Internal storage] in this manner force the robot to repeat the lesson learned in the future.

The use of microprocessors and programmed microcontrollers in systems for efficient utilization of fuel and energy resources is becoming very acute. Equalization of loads, switching off secondary loads at a given time with automatic switching on according to a program, switching off loads when energy consumption approaches the programmed value, uniform distribution of load during given periods and so on provide a decrease of energy consumption and conservation up to 20-30 percent.

The use of microcontrollers and other microprocessor systems yield a 20-30 percent saving of energy in institutions and housing due to correction of the operation of boilers, pumps, blowers, refrigerators, elevators and so on. Thus, for example, the use of microprocessors in the control equipment of heating-production boiler plants of group buildings permits a reduction of energy consumption by 30 percent. The program planned for 1981-1985 to convert devices and computer equipment to microprocessor equipment can be realized only with extensive use of the entire nomenclature of general-purpose microprocessor BIS and the use of a limited nomenclature of specialized microprocessor BIS and microassemblies developed and manufactured by Minpribor. A number of specialized BIS and specialized microprocessors which realize the functions of a device for communicating with the facility, integration with peripheral devices and with communications lines that regulate microcontrollers, sensors and signal converters, specialized measuring and analog microprocessors, specialized microprocessor BIS with error detection and correction and also BIS for electronic scientific and large crystal clocks, for associative memories and associative processors requires development and production at Minpribor.

These and other specialized microprocessor BIS should supplement standard sets of microprocessor BIS produced by Minelektropribor and should increase the efficiency of utilizing them. The need for specialized BIS, including microprocessor BIS, is determined primarily by developments of SM EVM, ASVT-PS and ChPU and ASPI devices. Efficient selection of the ratio of general-purpose microprocessor BIS and special-purpose microprocessor BIS and microassemblies is determined for each of the groups of devices and SVT developed at Minpribor on the basis of the technical and economic concepts which include reqirements on metrology, reliability, cost and so on.

Introduction of microprocessor equipment to perform the digital part of processing will determine the need for simultaneous improvement of processing the analog part of information, which comprises the basis of devices for communicating with the facility, and numerous measuring devices. Development of integration devices which link the analog and digital equipment is required for 30-40 percent of all the

microprocessors used. A number of highly accurate and precision analog-digital and digital-analog devices has not yet been realized in the form of monolithic integrated circuits and they are produced in the form of hybrid microcircuits and microassemblies.

Along with the significant success achieved, there is a number of serious deficiencies and difficulties which limit and delay extensive introduction of microprocessor equipment in instrument building and in control-computer equipment. The most important of them are the following.

1. The problem of development and production of automatic testers and other devices for automation of checking and testing of microprocessor sets of BIS and printed circuit cards based on them.

In fact, a continuous reduction of the cost of microprocessor sets of BIS is accompanied by a gradual increase in the cost of checking, testing and diagnosing their malfunctions. This occurs for the reason that continual complication of the structure of circuits and an increase of the degree of integration of microprocessor BIS requires the use of more complex automated testers and other test equipment for checking the functioning of BIS. On the other hand, the functional density of schematic printed circuit cards is being increased constantly, which requires more complex and expensive automated testers and other test equipment. The problem of development and creation of devices for automation of monitoring, testing and diagnosis of microprocessor BIS and saturated schematic printed circuit cards must be fundamentally solved in this regard during the 11th Five-Year Plan.

2. The problem of development and production of special microprocessor equipment and programs for automation of the model-building process, development and debugging of hardware and software equipment of microcomputers and programmed microcontrollers and comparative analysis of different microprocessors.

The absence of serially produced devices among those named slows down development, programming and debugging of specific microprocessor systems and analysis of their characteristics.

Extensive research and experimental-design work to develop and assimilate both relatively inexpensive, but specialized (for some set of BIS) as well as more expensive, but more universal (for a wide nomenclature of microprocessor BIS) design, mockup and debugging systems of microprocessor systems and also simpler and less expensive (developers of microsystems) which provide automation of the process of preliminary and approximate analysis of the parameters and programming of different microprocessors and microcomputers must be organized during the 11th Five-Year Plan [6]. In many respects this system can replace a more expensive system for design and debuggin of microprocessor hardware and software.

3. The problem of microprocessor development, the structure and configuration of which are efficiently oriented or adjusted to specific user problems, to organization of multimicroprocessor control computer systems and so on.

The structure and configuration of serially produced domestic microprocessors are weakly oriented even toward mass user problems and toward operation in multimicroprocessor systems where mutual interlocking, synchronization and so on are important.

The reliability and viability of serially produced domestic microprocessors are inadequate. To increase their reliability, the structure and configuration of microprocessors should contain a means of dynamic rearrangement of the structure and special instruction sets, which will at the same time improve the efficiency of using the resources.

4. Weak coordination of work between developers of microprocessor sets of BIS and developers of control instruments and devices.

5. The absence of a complex program for development of monitoring and diagnostic and metrological equipment for debugging hardware and software.

6. The absence of BIS for effective realization of devices for communication with the facility, interfaces and peripheral devices.

Equipment developers are being provided with inadequate normative-technical documentation for microprocessor sets of BIS produced and being prepared for production.

Complex and purposeful programs for development of the required nomenclature of future BIS, SBIS, ZU, TsAP [Digital-analog converter], ATsP [Analog-digital converter], microprocessor sets of BIS and microcomputers for the ASM EVM, ASVT-PS, ChPU devices for measuring complexes and analytical and industrial devices and scientific instrument building must be developed for 1981-1985 for solving the indicated problems.

BIBLIOGRAPHY

1. Prangishvili, I. V., "Mikroprotsessory i mikro-EVM" [Microprocessors and Microcomputers], Moscow, Energiya, 1979.

2. Prangishvili, I. V. and G. G. Stetsyura, "Mikroprotsessornyye sistemy" [Microprocessor Systems], Moscow, Nauka, 1980.

3. "Development of an Adaptive Controller," ELEKTRONIKA, No 13, 1978.

4. CONTROL ENGINEERING, Vol 24, No 12, 1977.

5. Germans and Faruell, "A Controller of a Single-Structure Proce.s With Adaptive Amplification," ELEKTRONIKA, No 24, 1978.

6. Bailey, C. and T. Cachel, "General-Purpose Devices for Design of Microsystems," ELEKTRONIKA, No 18, 1979.

6521
CSO: 1863

NETWORKS

ARCHITECTURE OF COMPUTER NETWORKS

Moscow ARKHITEKTURA VYCHISLITEL'NYKH SETEY in Russian 1980 (signed to press 24 Oct 80) pp 4-6, 227-233, 252-253, 276-277

[Excerpts from book "Architecture of Computer Networks", by Eduard Aleksandrovich Yakubaytis, Izdatel'stvo "Statistika", 17,000 copies, 279 pages]

[Text] A new and rapidly developing field of engineering cybernetics, related to the development of large data acquisition, storage, processing and transmission systems, consisting of many computers, is examined in this book. Emphasis is placed on the latest type of architecture -- parcel switching computer networks.

The book is intended for managerial personnel, scientists, designers and developers, university students and for other people interested in problems of the development of modern computer networks.

Introduction

Electronic computers, which perform complex mathematical operations, quickly evolved into general purpose data processing systems. The development of large and superlarge memory systems led to the appearance of electronic libraries, in which the bodies of knowledge that have been accumulated through the ages, are stored quickly.

The transformation of telephone and telegraph networks into information transmission networks, and the appearance of special communications networks for computers and terminals, reflect the extensive capabilities for exchanging information over great distances.

Increasingly sophisticated multicomputer associations, which perform information acquisition, storage, processing, transmission and delivery, have appeared as a result of the development of methods of interacting with computers, and also as a result of the most efficient utilization of their capabilities. The development of these associations heralds [3, 17, 18, 27, 41] a qualitative leap in the development and application of computer technology, electronic libraries and communications technology. In the next few years multimachine associations not only will exert a significant effect on the development of science and various sectors of the national economy, but they also will play an enormous role in processes associated with training, education, social relations and other social

aspects of modern society. An examination of one of the most important classes of multicomputer associations -- computer networks -- is the subject of this book. In the next few years computer networks not only will determine the future of the information processing industry, but they will also acquire great social significance, affording large communities of the population an opportunity to obtain information about political events, news, transportation work, social services, etc. Local computer networks will make it possible to elevate the management of industrial associations, industrial and agricultural enterprises to a qualitatively new level. The application of network architecture in the development of single-junction time sharing computer centers will afford many customers an opportunity to gain access to most diversified computer information resources.

Today, unfortunately, there still is no generally accepted terminology on multicomputer associations. Therefore a glossary of the terms used herein is given at the end of the book. Abbreviations and conventional symbols are defined and explained in a list inserted at the end of the book, where there is also a subject index, so that the book can be used as a reference manual.

7.8. EVS Computer Network

The development at the LaSSR Academy of Sciences of the experimental computer network (EVS) is aimed at two main objectives:

scientific research in the field of computer network architecture;

development of a base for a general academic scientific research automation system.

The EVS architecture is determined by the following basic requirements:

the network does not have a main (central) computer. Any subscriber machine, depending on the priority assigned to it or the priority of its task, can interrupt the operation of another subscriber machine and assign it a task;

the network is random access, i.e., if a connection with the communications network is broken any subscriber machine can operate independently within its own capabilities;

the subscriber machines in the network are specialized in terms of structure and software to perform certain groups of functions;

the network utilizes conveyor information processing, in which different computers and machines participate;

macro-, mini- and microcomputers can operate in the system with different operations systems;

the network has high viability and continues to function if any of its elements fails;

the network is a growing network, i.e., it has the capability of expansion and development;

the network offers customers an opportunity to use existing computers, terminal complexes and the applied software, developed for them;

different process interaction procedures can be used in the network;

working machines (host machines) are relieved as much as possible of auxiliary tasks.

In view of the experimental nature of EVS, one more requirement should be added to the ones examined above: the network should have the capability for a wide range of scientific research in accordance with the architecture of multicomputer associations.

The eight-computer experimental computer network was delivered to the scientists of three institutions in August 1977. By late 1979 the number of computers in these and other institutions in the network increased somewhat, but the network software changed to a greater extent and the services it provides were expanded.

The first installation of EVS is shown in Figure 7.16 as it existed at the end of 1979. The network consists of interacting working, dispatcher, terminal-interface, communications and terminal machines.

The working machines determine the basic computer information resources of EVS. Each of them (Figure 7.17) has an OS 4.1 operations system, KROS, KAMA or DUVZ systems program, and also sets of programs, called logic interface converters and transport station.

The KROS program performs computer operations planning:

improvement of computer productivity;

automation of several computer operator functions;

expansion of capabilities of computer (ordering of tasks, control of assignment stream, etc.);

reduction of resource losses;

capability of long-distance assignment input-output.

The KAMA program performs dialogue processing of information messages, offering the following:

a unified data base for clients' programs;

controlled long-distance access to data bases;

service systems for constructing information complexes.

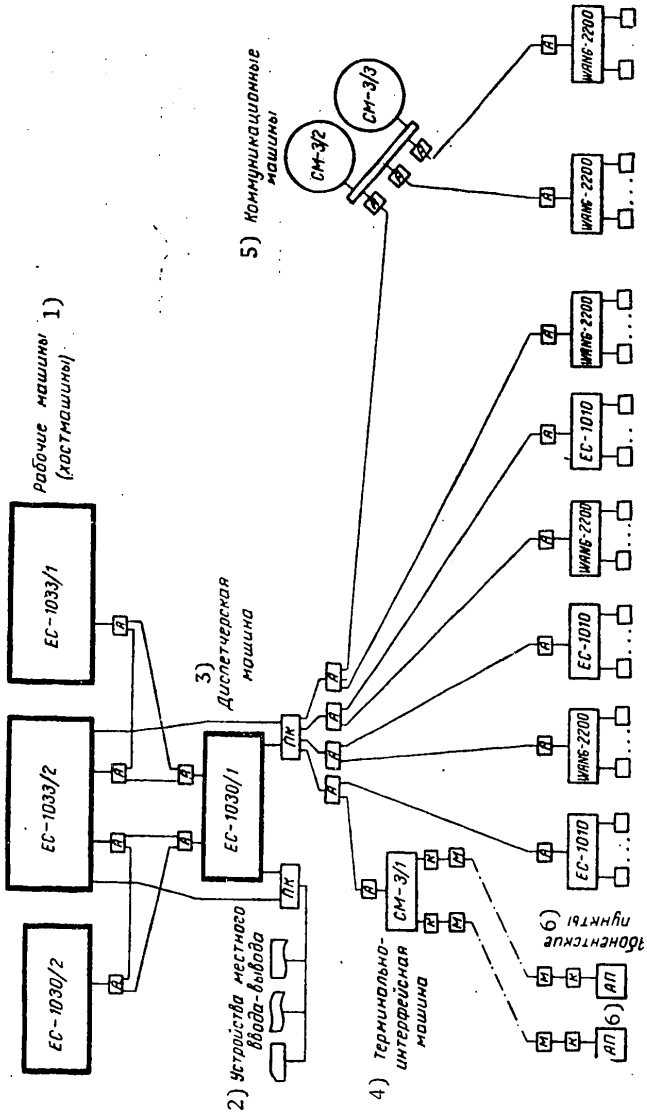


Figure 7.16. Experimental computer network of LaSSR Academy of Sciences.
Key: 1. Working machines (host machines)
2. Local input-output systems
3. Dispatcher machine
4. Terminal-interface machine
5. Communications machines
6. Subscriber stations

[EC=YeS; CM=SM]

The DUVZ program in the interaction mode performs:

preparation of tasks;

development of a data library;

long-distance task input-output;

program correction.

The logic interface converter enables processes that are being completed in YeS EVM [Unified series computer] to interact with the network.

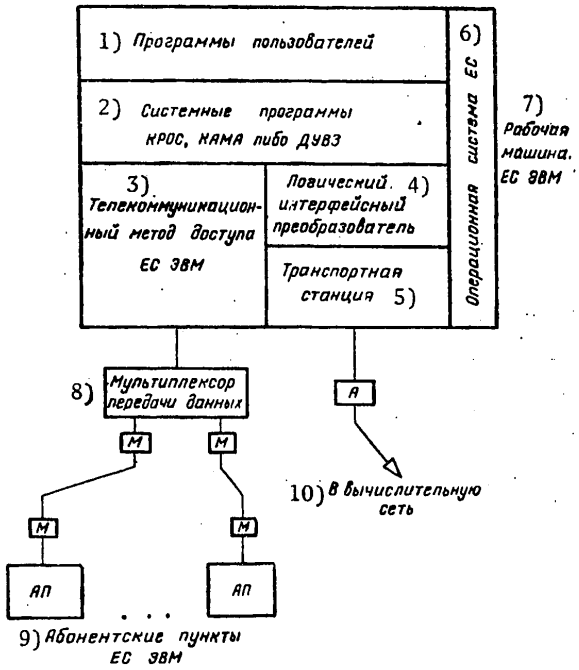


Figure 7.17. Program structure of working machine (host machine).

1. Client programs
2. Systems programs KROS, KAMA or DUVZ
3. Telecommunications access for YeS EVM
4. Logic interface converter
5. Transport station
6. YeS operations system
7. YeS EVM working machine
8. Data transmission multiplexer
9. YeS EVM subscriber terminals
10. To computer network

The transport station controls the connection of these processes through EVS logic channels.

The dispatcher machine contains a DISPATCHER 1.6 program package, which performs the following functions:

controls data streams passing through local input-output systems (perforation and printing systems);

receives tasks from terminal machine users;

converts formats and codes;

monitors received information and requests retransmission in the event of an error;

bufferizes messages in immediate access memory on magnetic disks;

selects the working machine that will complete the next task and transmits tasks to it;

stores tasks and the results of the completion of tasks in immediate access memory;

transmits the results to terminal machine users in the urgent or immediate delivery modes;

stores client files on disks of the dispatcher machine and transmits them on request to clients;

collects statistical information on the performance of the computer network;

distributes tasks among several working machines, completing computer operations (KROS).

The dispatcher (see Figure 7.16) is a YeS-1030/1 computer. During repairs, or in the event of an emergency, its functions are transferred to a YeS-1033/2 computer.

The SM-3/2 and SM-3/3 (main and backup) communications machines back each other up for reliability and perform the following functions:

parcel routing;

checks parcels after transmission on each channel that connects a pair of machines;

temporary storage of parcels in immediate access memory.

The dispatcher and switching machines (Figure 7.18) establish in EVS logic channels, which connect the processes that are being performed in working and terminal machines. This enables any terminal or any subscriber station to interact with all the systems programs (KROS, KAMA, DUVZ) that exist in the network.

The terminal-interface machine (SM-3/1) enables standard YeS EVM subscriber stations to interact with working and terminal machines via telephone channels.

The terminal machines function in accordance with the TISA-1010 or TISA-2200 program packages, which are run in YeS-1010 or WANG-2200 minimachines, respectively. Each of these packages performs interactive task preparation, text library management and interaction with YeS EVM systems programs (KROS, KAMA and DUVZ).

Information is exchanged between working and terminal machines by Dannyye packages in accordance with Recommendation X.25, approved by the International Telegraph and Telephone Consultative Committee. For this purpose logic channels, which connect terminals or subscriber stations with the processes which they need, are selected in EVS (see Figure 7.18). Machine interaction in EVS is described by an hierarchy of procedures, which includes seven levels: physical, channel, network, transport, session, representative and applied.

The physical channels in the network are parallel (the continuous lines in Figure 7.16), or sequential (dot-dash lines). Parallel channels are generated in groups, each with 28 twisted telephone cable pairs. Machines are connected to these channels by adapters (A), which perform parallel (nine-bit) symmetric half-duplex asynchronous exchange of parcels between machines at speeds of 0.2 (for a distance of 2,000 m) to 1.5M bits/s (for a distance of 50 m). The telephone network is used for sequential (bit-wise) communications. Information is transmitted on these channels at speeds of up to 2,400 bits/s. All channels are clear for all types of texts and coding methods.

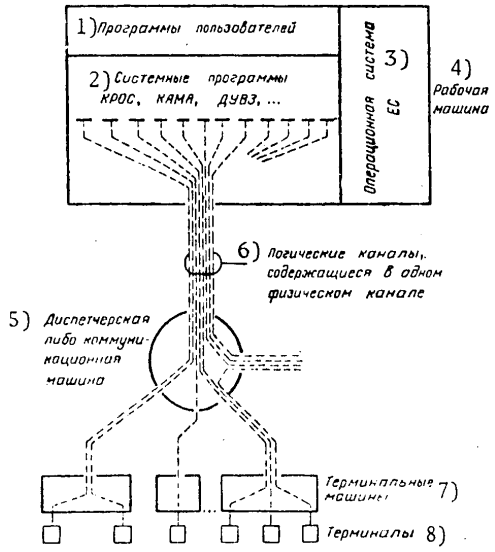


Figure 7.18. Logic channels in EVS.

Key: 1. Client programs
2. System programs KROS, KAMA, DUVZ, ...
3. YeS operations system
4. Working machine
5. Dispatcher or communications machine
6. Logic channels contained in one cable channel
7. Terminal machines
8. Terminals

Services Offered by Network. EVS is a continuously expanding computer network, and the services offered to clients are being expanded at an ever increasing rate. The computer network offered the following services in late 1979:

creation of text libraries in terminal machines;

text editing in dialogue mode;

dialogue preparation in terminal machines of tasks in YeS EVM languages (PL/1, Fortran, Assembler, etc.);

management of local task banks and solutions banks;

computer operations planning;

storage of any text in data base;

completely automated long-distance input of tasks and delivery of solutions;

makes available to a client a selected machine;

addressless task completion;

controlled remote access to data bases and information complexes;

"post restante" storage of solutions;

automatic recoding and reformatting of information for transmission from one subscriber machine to another;

automatic transcription of information from the carrier of one machine to the carrier of another (for instance from punch cards of a YeS EVM computer to a magnetic disk of a minicomputer);

transmission of inquiries about the passage of tasks and solutions through the computer network;

exchange of information (electronic mail) between operators;

transmission of information to external systems of various subscriber machines.

Control of Computer Network. EVS is controlled from the control center of the computer network. The control center is equipped with a group of displays, which duplicate input and printout information of the operator consoles of working and dispatcher machines. The network operator (the mathematician who controls the network) monitors the operators of the indicated machines and corrects their work when necessary. If a computer, channel or any other system fails the network operator disconnects it, guaranteeing that the rest of EVS will operate stably. The computer network operates around the clock.

Statistics are compiled in EVS on its qualitative and quantitative performance characteristics. The most important accounting parameter is the network accessibility coefficient

$$K_{na} = T/1440,$$

where T is the time allotted to a client for working with the network (the network is in good working order), min; 1440 is the number of minutes in a day.

The work time of each central processor of all working machines is also logged in EVS. Parcel streams through cable channels are also recorded (in each individual direction). Necessary inquiries about the performance statistics of EVS from the start of consecutive days are transmitted on request at any desired time.

Conclusions

The performance of parcel switching computer networks demonstrates their high efficiency, the possibility of offering clients access to numerous and diverse resources and means of interacting with them, including dialogue modes with many computers. Computer networks have proved to be so important that their development in terms of their significance is often compared with the very development (in the past) of computer technology.

A computer network combines into a single process all diverse information processing processes: control of technological conditions, completion of scientific, planning and financial calculations, storage of documents and data retrieval. By virtue of this it is possible to turn over all jobs related to the acquisition, storage, transmission, retrieval and delivery of information to industrial rails. There is now a new, extremely important sector of the national economy.

The effectiveness of a computer network depends directly on its size. Therefore the development of many independent far-reaching networks, belonging to different ministries and departments, will not provide all the positive features that networks offer. On the one hand, the development of an extremely large homogeneous computer network involves unnecessary capital investments, inconveniences and unreliable performance. The alternative is a multilevel hierarchy of compatible computer networks. In this hierarchy the computers and cable channels are divided by significance into several categories and are placed accordingly on different levels. The lower the level of a network the smaller its optimum dimensions should be.

Special importance is attached to the standardization of the parameters and characteristics of computer networks. There is a great danger that the manufacturers and users will spend many millions of rubles on the development of good networks, which will be very similar to each other, but which will end up being incompatible.

It is noteworthy that the conception of computer network hierarchy makes it possible in the very first step (if the necessary standards are maintained) to build local networks, not connected to any other network. And they need not

necessarily be large ones. Only 5-7 minimachines, specialized to perform certain functions, are necessary for developing a first-level computer network.

Computer networks can be built (and are being built) on the basis of ordinary telephone communications channels. However, their economic indices and performance characteristics are improved considerably when they are converted to digital techniques. Therefore the development and construction of a national communications network, based on the use of high-speed (more than one megabyte per second) digital channels and microcomputer communications control technology, is a very important problem.

Computer networks are beginning to find ever increasing application for the acquisition, storage and transmission of mass information. In the coming years they will include digital television, "electronic newspapers" and electronic mail. Therefore computer networks with each passing year will acquire ever greater social importance and will enable millions of clients to exchange information.

Contents

7872
CSO: 1863

COMPUTER NETWORK ANALYSIS AND SYNTHESIS METHODS

Leningrad METODY ANALIZA I SINTEZA SETEY EVM in Russian 1980 (signed to press 31 Jul 80) pp 2-6, 94-95

[Annotation, table of contents, and foreword from book "Computer Network Analysis and Synthesis Methods", by Gennadiy Fedorovich Yanbykh and Boris Yakovlevich Ettinger, Izdatel'stvo "Energiya", 7,000 copies, 96 pages]

[Text] The problem of developing optimally organized computer networks for hierarchical branch automated control systems is examined. The solution to the overall problem is reached through successive computations carried out with models of the particular problems involved. New results from research on the function and synthesis of computer networks are presented. The presented methods can be used to determine the number and distribution of computer centers and the configuration of computer complexes and data transmission networks.

The book is intended for the planners of large information and control systems, as well as for graduate students and students of the appropriate specialties.

Contents

Foreword

Integration of data transmission resources and computer hardware, which began in the 1960's, is now one of the dominant factors of scientific-technical progress. The consequences anticipated from arriving at a practical solution to the problem of shared use of information and its processing resources are often compared with the results of creating systems for transmission and shared use of energy, which led to a technical revolution in all areas of human endeavor.

Communication technology permits collective access of users (subscribers) to the computation capacities of remote computers and to data concentrated within their memories, exchange of data between remote computers, and unification of their output capacities. The aggregate of territorially dispersed computers, interacting long-distance between each other and with many subscribers through connecting telecommunication resources, is called a computer network.

Computer networks are being developed in our country on the basis of plans for creating a national automatic system for collecting, transmitting, and processing information, and territorial and branch automated control systems (ASU's).

The "Basic Directions for Development of the USSR National Economy in 1976-1980", adopted by the 25th CPSU Congress, state: "Promote further development and improvement of the effectiveness of automated control systems and computer centers, uniting them into a national information collecting and processing system in support of accounting, planning, and control. Create time-sharing computer centers" ((1), page 174).

This book examines computer networks intended for hierarchical (integrated) branch ASU's. An integrated branch ASU (OASU) is a hierarchical set of automated information and control systems (US's) intended to solve control problems at different

organizational levels of the given sector. Thus the OASU of the air transportation sector, which has a trilevel organizational structure, includes the following basic US groups:

High-level US's servicing the administrative machinery of the ministry and all-union associations;

middle-level US's servicing the administrative machinery of the sector's territorial production associations;

low-level US's servicing the administrative machinery of enterprises;

operational US's, particularly the automated airplane ticket sales and reservation system.

Because the activities of different sectors are spread out over large territories, because the volumes of information to be transmitted and processed are large, and because the requirements on the efficiency and reliability of information handling are high, branch computer networks must be used as the technical foundation of OASU's.

Development of a branch computer network requires solution of numerous complex scientific-technical problems, to include that of synthesizing the optimum structure of the network. The structure of a computer network depends on the composition and distribution of its hardware, including the communication channels existing between territorially dispersed hardware complexes intended for collection, transmission, accumulation, processing, distribution, and read-out of data. An optimally organized branch computer network must support execution of a given list of tasks assigned to the control systems of the OASU, within certain limitations imposed by the time and precision of data transformation, and with minimum outlays on creating and operating the network.

Because creation of a branch computer network requires outlays totaling tens of millions of rubles, the importance of optimizing its structure cannot be doubted. An exhaustive solution to this problem cannot be found in this country's literature, or in available foreign literature. Only certain problems associated with developing elements and subsystems or with selecting certain structural parameters of computer networks have been examined to date. One of the possible approaches to synthesizing the configuration of a computer network in application to the particular features of a state automated control system (OGAS) is presented in general terms in (2). However, it does not examine the problems associated with limitations imposed on traffic capacity and on the reliability of data transformation in the course of computer network synthesis, and only approximate methods are offered for solution of synthesis problems. Somewhat earlier, the authors published a book (3) discussing the basic elements of an integrated methodology for synthesizing optimally organized branch computer networks. Since the time it was written, however, new results have been obtained, a significant part of which pertain to problems not examined in (3). In particular the following have been developed on the basis of the systems approach: the statement of the general problem of synthesizing the optimum structure of a branch computer network, the methods for breaking down the problem into a set of mutually associated particular problems associated with analyzing the traffic capacity of hardware complexes and optimizing the structural parameters of the

network, and an iterative flowchart of calculations used in designing the structure of a branch computer network, based on models simulating particular problems. New methods based on systematic application of models of multiphasal queuing systems, models of discrete mathematical programming, and optimization algorithms employing the "branches and boundaries" method are presented as the means for solving the particular problems.

These results are generally reflected in this book.

This book is methodological in nature. Unfortunately, due to limited space it was impossible to completely explain all aspects of the engineering computations required in the synthesis of the structure of a branch computer network. Only the basic idea of such computations is described, and publications examining the computation techniques and offering numerical results are cited. For the same reason we were unable to review all known works pertaining to computer network synthesis.

The book consists of five chapters. Chapter One presents the methodology of synthesizing the structure of a branch computer network on the basis of a hierarchical system of models. Chapter Two describes models simulating the function of hardware complexes and subsystems in a branch computer network. Chapter Three is devoted to solving the problem of optimum distribution of computer centers and assignment of subscribers to them. Chapter Four describes the models and algorithms for synthesizing optimum configurations of computer multisystems intended to support computer centers. Chapter Five presents the methods of optimizing subscriber and intercenter communication networks.

This book presents the results of research conducted by the Central Scientific Research Institute of Civil Aviation Automated Control Systems (TsNII ASU GA). The authors obtained these results independently, and in collaboration with Candidate of Technical Sciences V. I. Bobr, Candidate of Technical Sciences B. A. Stolyarov, and V. F. Trigub.

G. F. Yanbykh wrote chapters Three and Five, and §11, and B. Ya. Ettinger wrote Chapter Two. The remaining material was written by the authors jointly.

Please send remarks and requests associated with this book to the following address: 191041, Leningrad, D-41, Marsovo Field, d. 1, Leningrad Division, Izdatel'stvo "Energiya".

The Authors

[138-11004]

11004
CSO: 1863

STRUCTURE, FUNCTION OF BRANCH COMPUTER NETWORK INTRODUCED

Leningrad METODY ANALIZA I SINTEZA SETEY EVM in Russian 1980 (signed to press 31 Jul 80) pp 7-10

[Excerpt from book "Computer Network Analysis and Synthesis Methods", by G. F. Yanbykh and B. Ya. Ettinger, Izdatel'stvo "Energiya", 7,000 copies, 96 pages]

[Text] Chapter One. The Problem of Synthesizing the Structure of a Branch Computer Network

1. Structure and Functional Hardware of a Branch Computer Network

A branch computer network consists of computer centers (VTs's), information terminals (IP's) and the data transmission network (SPD)--Figure 1.

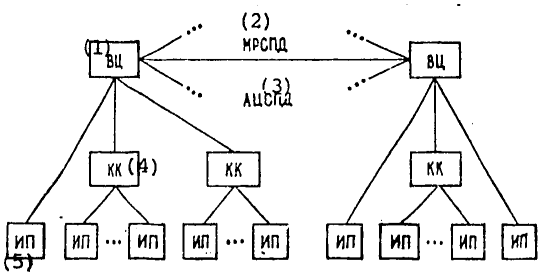


Figure 1. Structure of a Branch Computer Network

Key:
1. Computer center
2. Intercenter decentralized data transmission network
3. Subscriber centralized data transmission network
4. Multiplexing-concentrating center
5. Information terminal

The hardware of a VTs consists mainly of computer complexes (VK's) that process data in response to orders from the US [information and control system]. A computer complex is defined as a local (within one VTs) association of computers at the hardware and software levels. Modern series-produced computers are integrated together by one of the following methods, or by one of their combinations: 1) direct

connection of processing units, 2) collectivization of the main memory, 3) collectivization of external units, 4) interlinking of input-output channels. The second method is believed to be one of the most effective today (4). But the third and fourth methods are also extensively employed in relation to different kinds of VK's in which functions are clearly apportioned among computers of different types (4). A typical example of combined integration is the heterogeneous bilevel VK of the civil aviation branch computer network, described in (5). The lower level of this VK is represented by an ASVT M-7000 (SM-2) minicomputer with two processing units (and with a common main memory), which performs the function of interlinking the VK with communication channels and terminals. The upper level of the VK has one of the standard multisystem configurations based on YeS EVM hardware, and its functions include problem-oriented data processing in response to orders from the US. The computers at the upper and lower levels interact through interlinked input-output channels, or through immediate-access external storage units (an accumulator).

Outlying subscriber complexes (APK's) at the information terminals collect and record raw data, transmit it to the VTs for processing, and receive, display, and record results returning from the VTs.

Modern APK's are organized according to the aggregate-block principle. An aggregate system of peripheral hardware is a general-purpose set of functionally specialized blocks and units integrated with unified connections. These include, in particular, the aggregate systems of peripheral hardware of the SM and YeS EVM systems.

APK's now in existence and presently on the drawing board consist of one or several terminal units connected to a single group control unit. The terminal units are used to form the input data and (or) read out the output data in the language adopted for communication between subscribers and the computer network. The group control unit links the connected terminals to the communication channels (in the case of APK's communicating directly with a remote computer) or it performs the function of centralized preparation of data on input-output media (in the case of APK's operating independently). In cases where APK's possess their own computation capacities (for example when minicomputers are used as the group control unit), the latter are used for intermediate data processing: Information is transcribed from one medium to another, entries are multiplexed, computations requiring relatively simple algorithms are made, and so on. Intermediate processing raises the effectiveness with which the principal hardware of the VTs is employed.

An SPD consists of two parts--the subscriber centralized data transmission network (ATsSPD) and the intercenter decentralized data transmission network (MRSPD). The ATsSPD supports long-distance exchange of information between an IP and the VTs. An MRSPD provides telecommunication between VTs's. The SPD is made up of communication channels (together with data transmission hardware) and information flow multiplexing-concentrating terminals (centers) (KK). Information flows are concentrated mainly in an ATsSPD, inasmuch as there is a considerable difference in this system between the operating speeds of APK's and a VTs, and because the flow of input (output) data to the VTs from subscribers (from the VTs to subscribers) is highly irregular. In the general case, concentration (multiplexing) and, correspondingly, demultiplexing in an ATsSPD proceed in several steps, at the level of

bits, symbols, blocks, and entire messages. Special frequency and time multiplexing apparatus and programmable long-distance concentrators based on minicomputers support the concentration functions.

So-called linking (communicating) processing units, consisting of multisystem configurations of minicomputers, are used to connect the main computer complex installed at the computer center with the communication channels of modern computer networks. The linking processing unit is responsible for message collection, temporary storage, input control, and editing, for switching and routing information flows, and so on. The linking processing unit is connected to the main computer complex on the basis of the computer integration method listed above.

The most effective method for organizing telecommunication between VTs's is now believed to be the multiplexing of messages and job batches (6). In this case the linking processors multiplex the messages. In the course of their interaction, neighboring VTs's (connected by direct MRSPD channels) exchange information directly. VTs's that do not have direct communication exchange their information by way of transit routes passing through neighboring VTs's.

The operation of a computer network is interpreted as a process of interaction between the structural and functional elements enumerated above. In the general case, this interaction boils down to exchange of information messages or signals, which may be interpreted as certain inquiries and the appropriate responses. The set of inquiries transmitted from subscribers may be subdivided into two categories. Inquiries in the first category are called simple. Simple inquiries from a concrete subscriber are serviced with the help of information pertaining to just the subscriber alone (information characterizing the status of that subscriber's production processes). Inquiries in the second category are called complex. In addition to information pertaining directly to the subscriber who transmits a complex inquiry, the servicing of a complex inquiry requires information on other subscribers as well.

These two situations may be illustrated with the example of selling airplane tickets for direct and connecting flights respectively. To sell tickets for direct flights, we would need information on the space available aboard airplanes at the airport of origin. When we process a request for a connecting flight, we would also need information on space available aboard the corresponding airplanes taking off from the transfer points (that is, from other airports).

The inquiries of each subscriber are recorded in the IP (with the help of an APK) and transmitted to the VTs to which the APK is connected by channels of the ATsSPD. The subset of subscribers in the computer network with their APK's connected to the given VTs is called the zone of this VTs. Inquiries (simple and complex) transmitted from subscribers in the zone to the zonal VTs are called primary inquiries. All primary inquiries are processed in the zonal VTs. In this case some of the primary inquiries, for the processing of which the zonal VTs possesses all of the necessary information (and programs), are serviced entirely by the zonal VTs. The reply to the inquiry is transmitted (when necessary) to the appropriate subscriber by the ATsSPD channels.

In the case where information absent from the given zonal VTs is required for the processing of a primary inquiry, that VTs transmits (after processing the primary inquiry) so-called secondary inquiries to other VTs's possessing the necessary information. Then the zonal VTs forms a reply to the primary inquiry on the basis of the replies it receives to its secondary inquiries, and the results of processing the primary inquiry. As in the first case, this reply is transmitted by ATsSPD channels to the appropriate IP, where it is documented (displayed) by the hardware of the APK.

While all information necessary for the processing of simple inquiries from subscribers within a zone is located in the zonal VTs, secondary inquiries are transmitted out by the given VTs only when it receives complex inquiries from within its zone (or secondary inquiries from other VTs's) requiring information from subscribers in other zones for their complete processing.

[138-11004]

11004
CSO: 1863

NEW BOOK ON COMPUTERS AND COMPUTER NETWORKS

Moscow EVM I VYCHISLITEL'NYYE SETI in Russian 1980 (signed to press 20 May 80) pp 2-3, 9, 326-327

[Annotation, excerpt of introduction, and table of contents of book "Computers and Computer Networks" by Vasiliy Nikolayevich Kriushin, Inna Nikolayevna Buravtseva, Nina Mikhaylovna Pushkina, and Nina Grigor'yevna Chernyak, Izdatel'stvo "Statistika", 18,000 copies, 328 pages]

[Excerpts] Annotation

This textbook presents the principles of functioning of electronic computing machines and systems. The material is based on third-generation models of computers.

Attention is focused on logical organization of computing machines and systems, the characteristic features of encoding economic data, the structure of peripheral equipment, the characteristics of particular units, and their operating capabilities. The book reviews the principles of constructing collective-use computing networks, the equipment included in the networks, and user interaction with the computer.

The material in the textbook corresponds to the curriculum of the course "The Architecture of Computers and Computer Networks" for students at higher educational institutions studying in specialization No 1738. The book will be of interest to college students and specialists working in the field of computer applications in automated control systems and at computing centers.

Introduction

At the present time there are roughly 3,000 computing centers in the country [3]. At the start of 1977 more than 3,000 automated control systems were in use [2]. In 1978 alone more than 400 automated control systems for record keeping, planning, and management were set up [32]. More than 300,000 specialists in the country are working on setting up and running automated control systems [33].

Table of Contents

11,176
CSO: 1863

BASIC TELEGRAPH, TELEPHONE UNITS IN COMPUTING SYSTEMS REVIEWED

Moscow EVM I VYCHISLITEL'NYYE SETI in Russian 1980 (signed to press 20 May 80) pp 283-293

[Subchapter 11.3 of book "Computers and Computer Networks" by Vasiliy Nikolayevich Kriushin, Inna Nikolayevna Buravtseva, Nina Mikhaylovna Pushkina, and Nina Grigor'yevna Chernyak, Izdatel'stvo "Statistika", 18,000 copies, 328 pages]

[Text] The enormous importance of means of communication was pointed out by V. I. Lenin, who in 1918 wrote that "Socialism without the mails, telegraph, and machinery is an empty phrase" ("Poln. Sobr. Soch." [Complete Works], Vol 27, p 278).

The choice of means of communication depends on a large number of factors: the number of subscribers, the speed and reliability of transmission, the length of the communications channels, and the like. Telegraphic means of communication are most efficient in those cases which require high reliability of transmission; telephone communication is preferable for higher speed, and means of facsimile communication are best for transmitting drawings and figures.

The basic equipment used in telegraphy comprises various types of telegraphs. The contemporary models use the modified international telegraph code ITC-2 recommended for use by the International Telegraph and Telephone Consultative Committee. This code is a uniform, five-element code in which 0 transmits an absence of current and 1 the presence of current. For this reason all electrical elements are divided accordingly into elements under current and those which are not.

Each combination transmitted by a telegraph channel in ITC-2 code has a length of $5t_o$ where t_o is the length of a single signal. When telegraphs work in the start-stop mode, however, two additional service elements are introduced. The first one, the start, is without current and is transmitted before the code combination. It also has a length of t_o. The second element, the stop which completes transmission of the code combination and has the current position "1," is longer, 1.5-2.0 t_o. Thus, the total length of one code combination transmitted by telegraph channels in ITC-2 code is $7.5t_o$.

In addition to the general characteristics described above, various other indicators are used to define telegraphs, for example:

a. correcting capability, which characterizes the quality of work of the telegraph receiver and is defined by the magnitude of maximum marginal distortions with which it is still possible to record an error-free signal. This capability may be theoretical, effective (that is, measured under real operating conditions), or nominal (the lowest value for the group of similar devices);

b. theoretical productivity, which is the maximum number of words which can be transmitted and received by the telegraph in an hour;

c. technical productivity, which takes into account the time of transmitting useful information and service information;

d. operating productivity, which reflects the design characteristics of the equipment, operator qualifications, and the like.

All telegraphs are basically similar in design and have the following primary functional parts:

a. the transmitting unit, which feeds and converts the information to be transmitted. It comprises a keyboard, a coding device with five steel bars each of which may occupy one of two fixed positions corresponding to "1" and "0", by which the coding device carries out the coding; the transmitting distributor, whose primary purpose is to convert parallel code combinations formed by the bars into a sequence of electrical signals. The frequency of rotation of the distributors, which is a fixed quantity in all telegraphs, determines the length of a single signal t_o;

b. the receiving unit, whose functions are to receive, convert, and print the information received from communications channels on paper tape. The receiver consists of a distributor which performs the inverse conversions relative to the transmission, the decoder with five decoding bars, and a printer;

c. a drive mechanism;

d. auxiliary assemblies and units, including mechanisms to pull the paper and inking tapes, the register switch mechanism, a self-locking device, and others.

Telegraph lines use the domestically produced ST-35, ST-2M, STA-2M, RTA-6, STA-M67, RTA-7 and other telegraphs plus the East German T-51 and T-63 units.

The ST-2M telegraph, which is a modification of the ST-35, uses ITC-2 code. It has a speed of 50 bauds and a maximum operating range of 550 kilometers. The letter "A" in the STA-2M telegraph means "automated" because a transmitter attachment and a reperforator attachment are connected to it and allow it to be used for transmitting a premade punched tape and for receiving information from the channels on both paper tape and punched tape.

The availability of these attachments leads to fuller use of communications channels, and the possibility of preparing punched tapes in advance makes it possible to improve their quality. Furthermore, the use of punched tapes allows automation of the operation of retransmission of messages at central telegraph stations and makes it possible to use punched tapes as data media for computers.

The STA-M67 telegraph differs only slightly from the STA-2M.

The page-printing start-stop telegraphs were a further development of telegraphy technology. The RTA-6 page-printing telegraph with transmitter and reperforator works at a speed of 50 and 75 bauds, has a roll of paper 215 millimeters wide, and can receive three copies of a text simultaneously.

The RTA-7 page-printing telegraph, unlike the electromechanical devices considered above, is classified as an electronic-mechanical telegraph. In it certain mechanical assemblies such as the distributor, coder, decoder, and a few others are replaced by electronic devices. The RTA-7 works at speeds of 75 and 100 bauds and has three registers, automated attachments, a filter for noise suppression, a radio receiver, and a number of other auxiliary units.

Telegraph communication may be organized using either general-use communications included in the system of the USSR Ministry of Communications and serving various institutions of the population through a network of communications departments or by means of subscriber telegraphy, which serves those institutions that have telegraphs connected to the subscriber telegraph station.

One of the varieties of subscriber telegraphs is the international Telex telegraph, designed for transmitting messages to other countries.

Facsimile communications apparatus is used to transmit images (documents, photographs, drawings, tables, textual material, and the like) through communications channels. According to the definitions of the International Telegraph and Telephone Consultative Committee, the terms "phototelegraph" and "phototelegraphic apparatus" should refer only to the equipment used to transmit and receive half-tone images. The more general term is "facsimile apparatus" (a facsimile is an exact reproduction at the receiving point of the image being transmitted). This term covers devices that transmit both half-tone and hatchured originals.

There is a fairly broad range of facsimile communications equipment including the following devices:

FTA-PM, Aragvi, and Shtrikh — these devices are used only for receiving and transmitting hatchured images 220 x 150 millimeters (220 x 300 millimeters in the FTA-PM); they have transmission times, respectively, of 12.5, 6, and 2.1 minutes; the image is printed on ordinary paper;

Ladoga — designed to transmit hydrometeorological maps of unlimited length and up to 480 millimeters in width. It takes 22 minutes to copy a map 480 x 690 millimeters;

Neva — this device is used for work with both half-tone and hatchured originals. It takes six minutes to transmit an original that is 220 x 300 millimeters;

Gazeta-1, Gazeta-2 — used for transmitting newspapers; for this reason they are larger than the original (420 x 610 millimeters), can receive copies on photographic film, and have a more rigid requirement for misalignment of images, not more than one millimeter for each 100 millimeters of length of the paper. The devices can transmit one newspaper page in 50 and 2.3 minutes respectively.

All facsimile devices consist of transmitting and receiving parts. The transmitting part includes: a scanning device for breaking the image of the original down into individual elements; a light optical system by means of which a light beam is moved across the surface of the original secured to a plane or a revolving cylinder (depending on this a distinction is made between devices with planar or cylinder scanning); a photoelectric convertor that converts the light reflected from the original into electrical pulses whose magnitude depends on the brightness of the image; a video signal convertor; phase and synchronization devices, and the like.

The basic units of the receiving part of the facsimile apparatus are the video signal convertor, the writing and scanning devices, and the phase and synchronization units. They convert electrical pulses arriving from the communications channel into a stream of light which is then projected onto the surface of photographic paper (this method is called the closed method because the image becomes visible only after the photographic paper is properly processed) or onto electrochemical, electrothermal, and other types of paper. This method is called open because the copy appears immediately upon receipt).

Among the traditional means of communication are varieties of telephone equipment, including the TA-60, TA-65, and TA-72 (Soviet Union), TsB 621/65 andTsB 631/65 (Poland), V = 63 St (East Germany), TA-66a (Czechoslovakia), and other telephones, the ATS-47, ATS-54, and ATSK automatic telephone exchanges, and others.

Among the various means of communication, the most attention is devoted to building and using data transmission equipment (APD). This usually includes a device to protect against errors (UPZ), which feeds control signs to the code

combinations to protect the data being transmitted against errors; a memory unit whose capacity must be sufficient to store at least one code combination; a control unit which provides for interaction among all assemblies of the data transmission equipment; input-output devices and a signal conversion unit (UPS).

The signal conversion unit is used because the data to be processed by computer are written in binary form and represented by rectangular pulses. No additional operations are ordinarily required to transmit them by telegraph channels. But to transmit them by telephone channels the data must be appropriately converted. This is done using what is called the carrying frequency. The carrying frequency is selected in the middle of the pass band and the data is put on it. This process, called modulation, greatly increases transmission speed. Modulation may be done by frequency (the frequency of the carrying oscillation is subject to change), amplitude, or phase.

Modulation is done in special devices called modems which generate the needed sine curve and modify it according to data received. Demodulation of the carrying frequency also occurs in modems.

Some of the data transmission units used in practice are the Akkord-50, the Akkord-1200, and the Minsk-1500. The Akkord-50 data transmission unit (see Figure 11.4 below) works in a semi-duplex mode that provides data transmission by telegraph channels at speeds of 50 and 100 bauds. Transmission reliability is $3 \cdot 10^{-7}$. Data is fed from punched tape by an FS-1500 (Czechoslovakia) photoreader, and outputted to punched tape by a PL-150 tape punch that is connected to the Akkord-50 by a remote interconnection, punch, and reading unit. Data input and output may also be done by page-printing telegraph. Connections are made using the call device included in the Akkord-50 set. Cyclical coding and decision feedback are used to insure good-quality transmission.

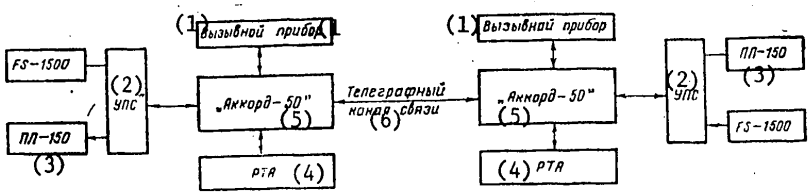


Figure 11.4. Structural Diagram of Data Transmission Using the Akkord-50 Equipment

Key: (1) Call Device;
(2) UPS [Signal Conversion Unit];
(3) PL-150 [Tape Punch];
(4) RTA [Page-Printing Telegraph];
(5) Akkord-50;
(6) Telegraph Channel.

The Akkord-1200 data transmission equipment is used for work with telephone communications channels and functions in both the semi-duplex (switchable channels) and duplex (assigned channels) modes at speeds of 600 and 1,200 bauds. High reliability of data transmission (no worse than 10^{-6}) is achieved by using cyclical codes with the polynomial $x^{16} + x^{12} + x^5 + 1$ and decision feedback. The Akkord-1200 works in the following modes: "telephone," which is designed for service communications; "data transmission," which only transmits data; "data reception," which only receives data; and, "internal," which is used to check the working condition of the equipment.

The Akkord-1200 set of equipment includes a PL-150 tape punch, an FS-1500 photo reader, a calling and ringing device, and an Akkord-1200 PP receiving-transmitting unit. If the subscriber point does nothing but transmit data, it will have an Akkord-1200 PD transmitter which works in three modes: telephone, data transmission, and internal, instead of the Akkord-1200 PP. The transmitter has a coding device that forms the check combination of the cyclical code; a data input control unit that converts parallel code into sequential, copies data into the memory unit, and forms the service characters of the data block; a phase device to determine the beginning point of phasing and establish phasing regimes (the synchronous method of transmission is used); devices to form control pulses and exchange signals, and the like.

On the other hand, if the subscriber point only receives data, the set of equipment includes only an Akkord-1200 PM receiver which works in the modes: telephone, data receiving, and internal. The receiver includes a decoding unit that detects errors in the block of data received; a phasing unit; an input register designed to convert data from sequential to parallel code; a device to form control pulses and feedback channel signals, and other equipment.

The Akkord-1200 has a memory unit for simultaneous storage of two blocks of data and a Modem-1200 that converts discrete signals received from terminal equipment into frequency-modulated signals suitable for transmission by telephone channels and for inverse conversions.

The Minsk-1500 automatic transmission equipment also transmits by telephone channels, but it has the capability of being directly connected to the Minsk-22 computer. An inverse code is used in the equipment to protect against errors; the reliability of transmission is 10^{-6}.

The DFE-550 automatic transmission equipment (East German) is also used. It transmits data by telephone channels at speeds of 600 and 1,200 bauds and employs cyclical codes which provide a data transmission reliability of 10^{-6}.

The information taken from the punched tape goes to the memory unit, which is designed to store data until it has been checked at the receiving station, and to the coding unit where the data is converted from parallel to sequential code and cyclical coding operations are performed.

The data then goes to the modem where it is converted so that it can be transmitted by telephone channels.

When data is received from communications channels the inverse conversion takes place in the modem in order to output DC pulses to the equipment. Conversion of the information into parallel code is accomplished by a convertor that is included in the decoding unit. The data then proceeds to the memory unit, to that zone in which it was located on the transmitting side. Then the correctness of the transmission is checked and if it went properly a confirmation signal is sent to the transmitting party by the feedback channel. If the transmission was improper a signal to repeat is sent.

In view of their differences in data coding, speed of input-output from the computer and transmission by communications channels, shape of signals, and the like, to operate data transmission and computers together it is necessary to match their parameters. One of the procedures for this presupposes the use of punched tape, which insures a fuller load on communications channels because preparation of the punched tape may be done throughout the working day, while it is transmitted only at a strictly determined time. But with this technique instantaneous data processing is impossible. Therefore, another technique of matching the computer and communications channels is considered better. This requires the use of special interconnection devices such as the Minsk-1560. This device can connect up to 32 telex channels with a transmission speed of 50 bauds in each line to a Minsk-32 or Minsk-23 computer. This procedure uses fixed (unswitchable) telegraph lines (in this case the number of user stations cannot exceed 32) and connection by four telephone lines through Akkord-1200 or Minsk-1500 data transmission equipment.

The Minsk-1560 device also locks out all communications lines when there is a malfunction in the computer and any particular line while it is working or in the absence of lines. The Minsk-1560 includes line equipment, a memory unit, a control block, a calling device, a page-printing telegraph, and so on.

When data is being fed from telegraph channels it goes first to the individual line equipment in which the telegraph messages are converted into pulsed signals, and then to the memory unit and shift register where the sequential code is converted to parallel code. The data then goes through the exchange register to the magnetic core storage of the computer to the address indicated in the appropriate work register. A zone of the magnetic core storage and duty register corresponds to each of the 32 communications lines.

When data is being removed from the computer a parity check is made in the exchange register and it is converted from parallel code to sequential code in the shift register. The telegraph messages are formed by an electronic relay.

All input-output operations are coordinated by a special input-output subroutine put in the computer in advance.

In addition to the above-mentioned means of communication, communications systems and computing networks in practice must include various switching devices, from very simple electromechanical ones to switching centers that use electronic equipment.

The first message switching devices were developed for telegraphy, where a rotating 360 degree selector was used as a switching element to connec subscribers. In such a commutator the brush of the selector moved across a contact field and was connected with the subscriber line through a plug-type switch.

With the invention of the telephone and the spread of telephone communications manually operated switchboards appeared, followed by automatic ones, including step-type and 360 degree selector commutators. The further development of switching technology led to coordinate selectors and matrix-type relay systems. In recent years semiconductor elements have begun to be used in place of relays in the connecting assemblies, which has led to the establishment of electronic switching centers.

Figure 11.5 below presents a structural diagram of a communications switching center used in a computer network. In the general case this will include the following: a line equipment block; an input-output block; a block of connecting elements; storage; and, the control unit.

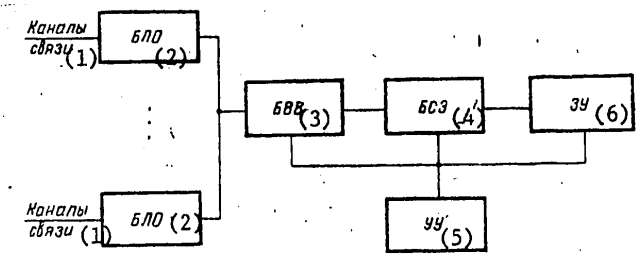


Figure 11.5. Structural Diagram of Communications Switching Center.

Key: (1) Communications Channels;
(2) Line Equipment Block;
(3) Input-Output Block;
(4) Block of Connecting Elements;
(5) Control Unit;
(6) Storage.

The line equipment block interconnects the communications channels by means of which remote subscribers are connected to the particular switching center. While performing this function the block constantly monitors the state of the communications channels so that when a request is received from a subscriber to set up communications the necessary line can be connected on time, the signal decoded correctly, and communications stopped

at the end of the transmission. Each block corresponds to a definite line and includes the equipment assigned to that line. Modems are installed in the line equipment block to convert input and output signals into forms suitable for internal movement at the switching center and for transmission by communications lines.

The input-output block is designed to recognize and appropriate signs (start and stop messages, tags for the beginning and end of messages, and the like) check to see that the equipment is connected correctly at the beginning, determine the address of the preceding switching center, and perform various other functions.

The block of connecting elements does the primary work of establishing communications between subscribers. This block is the most important of the unit at channel switching centers. The principal function of the connecting elements is to insure rapid, good communications between input and output subscriber lines. One of the methods of connection is separation in space (spatial switching, switching physically distinct circuits) where the coordination structure of the switching field is a matrix formed of points of intersection between input and output lines.

The configuration of the switching field may vary. Thus, the number of connecting junctions in a switching field constructed on a single-series non-interlocking scheme (see Figure 11.6a below) is equal to the product of the number of input and output lines. For systems with up to 1,000 inputs and outputs the most widespread system is the 3-series scheme (see Figure 11.6b below) which insures a minimum number of junctions with low probability of losing the call.

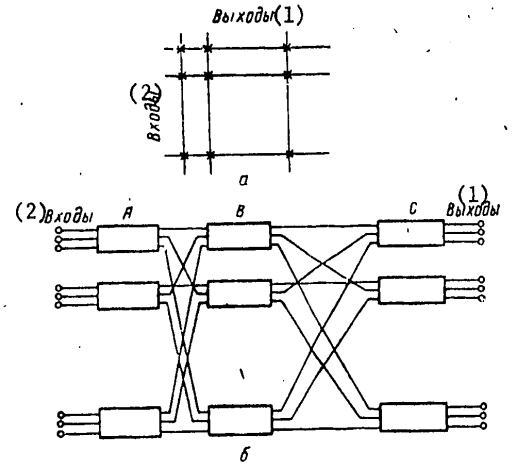


Figure 11.6. Switching Field Structures.

Key: (1) Outputs; (2) Inputs.

When designing a communications center an effort should be made to minimize the number of points through which the connecting path travels and insure that the probability of interlocking is insignificant.

The storage device may be small memory units in the form of registers (chiefly in the central switching channel) designed to store information for the time required to process the call and form the connecting route, to store data on bypass routes, and to store the various types of tables and references necessary to select routes and distribute messages. The storage of the switching centers may be organized large-capacity memory units; this is more typical of message switching centers. In this case several alternatives are possible for using memory. According to one two memory units are installed. One is located at the input of the message switching center and the other at its output. Upon receipt incoming messages are copied into the first memory unit until a sufficient number of them have been collected to organize switching or until the end of the message signal is received. After this they are recopied into the output memory units. Output to the communications channel is done either as the messages come in on the "first received, first served" principle or according to their priority.

Another alternative assumes the existence of one memory unit that stores all information. When a signal is received that indicates the necessity to switch to processing messages (check the correctness of receiving, extract service information, and the like), the output channel is checked to see if it is occupied. After all these operations are performed and the originating channel is free, the data is outputted to the communications channel. This procedure, which is the most popular today, found application in practice only after the appearance of magnetic tape, drum , and disk memory units.

The control unit coordinates the work of all the blocks of the switching center. Thus, the initial call sent by the subscriber is received by the control unit, which assigns the order of performance of operations by which the line equipment block will scan and identify the line on which the call came, analyze all possible directions of further message travel upon finding a free path, send commands to the block of connecting elements to make the required connection, and so on.

In computer networks communications processes, concentrators, and one or several computers may be connected to the switching centers. This equipment is used for partial message processing, route selection, and certain other operations.

The American IBM 5910 system may be cited as an example of a message switching center. Up to 480 telegraph channels can be connected to it through the line block 5974. The system has paired 5978 processors, one of which is in working condition and the other in reserve at any moment in time. The memory unit is composed of paired magnetic disks. The duplication of equipment included in this system provides high working reliability. With an average message length of 300 characters the switching device processes up to five messages a second. The message switching center is connected to the computer through a standard IBM selector channel.

[116-11,176]
11,176

CSO: 1863

BASIC CHARACTERISTICS OF YES REMOTE DATA TRANSMISSION SYSTEM REVIEWED

Moscow EVM I VYCHISLITEL'NYYE SETI in Russian (signed to press 20 May 80) pp 293-303

[Chapter 12 of book "Computers and Computer Networks" by Vasiliy Nikolayevich Kriushin, Inna Nikolayevna Buravtseva, Nina Mikhaylovna Pushkina, and Nina Grigor'yevna Chernyak, Izdatel'stvo "Statistika", 18,000 copies, 328 pages]

[Text] Chapter 12. Remote Data Processing Systems

12.1 Basic Information

Remote data processing systems, which are one of the foundations of computer networks, are used primarily:

a. for collection of data — data read from an intermediate medium at subscriber points (user stations) is transmitted to the computer or fed directly to the computer, eliminating intermediate recording. Data input operations at the user station are practically non-existent;

b. for outputting reference information — the computer processes a request received from a user station and sends the response to the station. In this case the volume of information being outputted usually exceeds the volume of information put in;

c. for solving problems of message switching — data is fed from one user station and outputted to another station virtually without processing;

d. for control of the computer when the user station is employed as a computer operator control console, and also for solving many other problems.

We will consider all the questions of remote processing using the example of the YeS [Unified System] computer remote data transmission system, which has already worked out systems such as YeS-7920, YeSTYeL, and others. This system includes various devices and all possible programs, standardized control algorithms and procedures, protocols that indicate the

principles of functioning of different elements of the system, and interfaces that define the conditions and parameters of interlinking remote data processing equipment.

The chief method of control in the YeS remote data processing system is the binary-synchronous method, which presupposes performance of the following procedures:

1. Establishing and maintaining synchronization in the data elements by means of synchronized symbols which may be included in any sequence of information symbols and also transmitted without them. These symbols are removed from messages received at the receiving station. Synchronization has three phases: the bit phase, which is intended to synchronize the signal conversion unit; the initial symbol phase, in which at least two synchronizing signals are sent — this must be done after establishing bit synchronization but before the beginning of transmission of each block of data; the phase of maintaining symbol synchronization, where two synchronizing symbols are added to the information stream each second during the process of data transmission;

2. Determining the readiness of the data element. After the synchronization procedure the transmitting and receiving stations exchange synchronizing symbols in each direction. At the appearance of the first binary combination that differs from this symbol, the synchronization "time-out" is switched on. Upon its completion either the synchronization procedure is done again or one of the three following procedures is performed: set the data link in a multipoint link, identify the station on switchable communications channels, or request receipt of data on assigned channels — data transmission in the primary mode. The last procedure begins after the data link is established. Transmission is done primarily by blocks using decision feedback.

3. Completion of data transmission. This procedure is performed when the signal for "end of transmission" is sent.

Figure 12.1 shows a block diagram of a simplified YeS remote data processing system. The interlinking devices in it are data transmission multiplexors which have connections, on the one hand, with the computer input-output channel and, on the other hand, with the user station through communications equipment. The data transmission multiplexor may be connected with the user station permanently by means of non-switchable (assigned) communications channels or temporarily using switchable channels.

The data transmission multiplexor may be connected with one user station (single-point connection) or several (multipoint connection).

The types of channels used and types of connections have a significant influence on the processes of establishing and breaking contact between data transmission multiplexors and user stations. Thus, with multipoint connection on assigned channels the multiplexor contact with the user station is established by sending special polling signals in the communications channel. When working on switchable channels contact is established by dialing the numbers of the desired user stations or computers depending on who is initiating the connection.

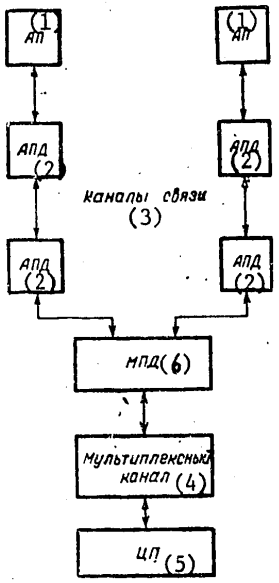


Figure 12.1. Block Diagram of Simplified Remote Data Processing System.

Key: (1) User Station;
(2) Data Transmission Unit;
(3) Communications Channels;
(4) Multiplex Channel;
(5) Central Processor;
(6) Data Transmission Multiplexor.

The YeS remote data processing system uses the five-element IPC-2 code for work on telegraph channels and the seven-element KOI-7 [possibly Data Processing Code No 7] developed on the basis of ITC-5 code for connections on telephone and broad-band communication channels. The translation of data from the internal machine codes KOI-8 and DKOI into transmission codes is done partly by software and partly by the hardware of the data processing multiplexor.

The YeS remote data processing system has two types of processing means: software and hardware.

All the remote data processing software interacts closely with the general computer software and works under its control.

The software, its structure, and the scope of functions performed depend on the configuration of the remote processing system and on the tasks it is performing. Moreover, they are oriented to particular models of computers and types of peripheral equipment.

Remote data processing software is subdivided into packages of applied programs that expand the capabilities of the operating systems and the basic software which facilitates the work of remote processing units and enables users to write programs in the symbolic programming language.

The principal components of the base software are the base telecommunications access method and the general telecommunications access method, which

is realizable in the OS [Operating System] and DOS [Disk Operating System] of the YeS computers and insures the establishment of communication with the user station, detection and correction of errors, control of buffer memory, conversion of data, and the like. It is more difficult to realize the base telecommunications access method because it requires that the user, when writing programs, have a detailed knowledge of the technical specifications and composition of equipment for data input-output at the user station and also be thoroughly familiar with methods and modes of data transmission. The user himself must include the addresses of user points and polling sequence in the programs and perform many other additional functions.

Because the general telecommunications access method virtually precludes any effect by the technical characteristics of the remote data processing hardware, this method provides a higher level of software support than the base method. It performs all the functions of the base method and, in addition, edits message titles, controls the organization of queues, establishes modes of interaction between the computer and the user station, and so on. The general telecommunications access method has been realized in the OS of the YeS system and is used chiefly in high-level remote data processing systems.

12.2. Remote Data Processing Hardware

The technical equipment used for remote data processing includes: data transmission equipment by which the data processing equipment is connected to the communications network; devices to interlink the computer and the data transmission equipment to insure control of data exchange, matching of electrical signals and speeds, and the like; user stations with various types of peripheral equipment. Figure 12.2 below provides a classification of the technical means used in the YeS remote data processing system.

The data transmission equipment makes it possible to work with both switchable and assigned telegraph and telephone channels as well as broad-band communications channels and physical lines.

Signal conversion devices, devices for protection against errors, and automatic call devices are included among the data transmission equipment of the YeS system, as well as other data transmission equipment.

The most widely used of these devices are signal conversion units. They are subdivided, in turn, into the following:

a. modems, which convert discrete binary signals from a computer or user station into signals that can be sent by telephone or broad-band communications channels, and perform the inverse conversion;

b. signal conversion devices for telegraph communications, used for work on telegraph lines to raise the level of the logical signal and its power;

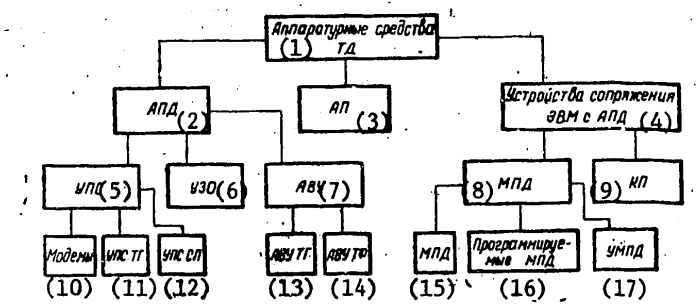


Figure 12.2. Classification of Technical Equipment for Remote Data Processing.

Key: (1) Remote Data Processing Hardware;
(2) Data Transmission Equipment;
(3) User Station;
(4) Devices for Interlinking the Computer and Data Transmission Equipment;
(5) Signal Conversion Devices;
(6) Error Protection Devices;
(7) Automatic Call Devices;
(8) Data Transmission Multiplexors;
(9) Communications Processors;
(10) Modems;
(11) Signal Conversion Devices for Telegraph Channels;
(12) Signal Conversion Devices for Connecting Lines;
(13) Automatic Call Devices for Telegraph Lines;
(14) Automatic Call Devices for Telephone Lines;
(15) Data Transmission Multiplexors;
(16) Programmable Data Transmission Multiplexors;
(17) Control of Remote Data Processing Multiplexors.

---

c. signal conversion devices for connecting lines, to transmit data over distances of up to 10-14 kilometers by connecting lines. The most widely used are low-level data conversion devices which convert discrete signals into low-level DC signals, which minimizes the mutual influence of signals of different circuits.

The technical characteristics of the various signal conversion devices developed for YeS remote data transmission systems are shown in Table 12.1 below. The devices for protection against errors code and decode messages and detect errors in them. They use matrix or cyclical codes which have the polynomials $x^{16} + x^{12} + x^5 + 1$ or $x^{24} + x^{23} + x^7 + x^5 + x^2 + 1$. Error protection devices with high-level codes are used in cases which require very high data transmission precision and communicate in only a few directions. The presence of a unit for protection against errors significantly increases the cost of building a remote data transmission system and makes

Table 12.1. Technical Characteristics of Signal Conversion Devices.

Model	Number	Manufacturing Country	Mode of Transmission	Type of Transmission	Transmission Speed, bauds	Type of Communications Channel	Modulation
Modem-200	YeS-8001	Soviet Union, Bulgaria	Duplex	Synchronous, asynchronous	200	Switchable and assigned telephone	Frequency
Modem-200	YeS-8002	East Germany, Hungary, Czechoslovakia	Duplex, semiduplex	Asynchronous	200	" "	"
Modem-1200	YeS-8005	Bulgaria	Semiduplex	Synchronous, asynchronous	600, 1,200	" "	"
Modem-2400	YeS-8010	Soviet Union	Duplex	Synchronous	600, 1,200 2,400	Assigned Telephone	Double Relative Phase
Modem-2400	YeS-8011	Hungary	Duplex, semiduplex	" "	1,200, 2,400	" "	"
Modem-4800	YeS-8015	Soviet Union	Duplex	" "	2,400, 4,800	" "	Triple Relative Phase
Modem-4800	YeS-8019	Soviet Union	Duplex	" "	24,000, 48,000	Broad-Band	Bipolar Amplitude
Low-Level Signal Conversion Unit	YeS-8027	Bulgaria	Duplex, semiduplex	Synchronous, asynchronous	50, 100, 200, 600, 1,200	Physical Line	-
Low-Level Signal Conversion Unit	YeS-8029	Soviet Union	"	" "	96,000	" "	-
Telegraph Sig-YeS-8030 nal Conversion Unit		Bulgaria	"	" "	50, 100, 200	Assigned Telegraph	-
Telegraph Sig-YeS-8032 nal Conversion Unit		Czechoslovakia	"	" "	"	Assigned and Switchable Telegraph	-

construction more complicated. In most cases, therefore, the necessary transmission reliability is achieved either by computer programs or by hardware and software of the user station. Table 12.2 below gives the technical characteristics of units for protection against errors [UZO's] related to the YeS remote data processing system.

The UZO-4800 is designed for work with user stations that are not included in the catalogue of the Unified System of Computers, while the UZO-48000 is for intermachine data exchange and the UZO-2400 for operations with AP-2 and AP-3 user stations.

The automatic call devices (AVU's) automate the establishment of connections in switchable networks and are subdivided into units designed for work on telegraph (YeS-8063) and telephone (YeS-8061 and YeS-8062) communications channels.

The following standards for unified interfaces have been established in the YeS remote data transmission system to standardize the interlinking of particular units: interface S1 for signal conversion devices with communications channels; interface S2 for signal conversion devices and automatic call devices with data processing equipment; interface S3 for error protection units with data processing.

Devices for interlinking computers and data transmission equipment consist of data transmission multiplexors (MPD's) and communications processors.

The data transmission multiplexors are the central units of the remote data processing system and determine its capabilities and configurations. These units provide:

a. interaction between the computer and the user station through communications channels, with performance of essential data conversion, partial buffering (the main computer memory is used for complete buffering), interference-stabilizing coding, and the like;

b. execution of data control algorithms realizable for each user station and containing procedures to establish and break the data linkup, for identification and inquiry, and for data transmission. The execution of the procedures is initiated in the data transmission multiplexor by a sequence of channel programs and commands;

c. control of data transmission channels and equipment; in other words, it insures logical and electrical interlinking of the computer and the data transmission equipment.

The YeS remote data transmission system presupposes several types of data transmission multiplexors that differ by the number and variety of pieces of data transmission equipment, user stations, and communications channels. In all data transmission multiplexors the computer is connected to the multiplex channel and the structure is similar, consisting of a unit to

Table 12.2. Technical Characteristics of Units for Protection Against Errors

Model	Unit Code	Manufacturing Country	Transmission Speed, bauds	Code	Method of Protection Against Errors	Mode of Transmission	Type of Communications Channel	Method of Connection to Channel
UZO-1200	YeS-8121	USSR	600, 1,200	ITC-5	Cyclical with Polynomial $x^{16}+x^{12}+x^5+1$	Semiduplex	Telephone	Through Modem-1200
UZO-2400	YeS-8122	Hungary	200, 600, 1,200, 1,400	KOI-7	Cyclical with Repetition	Duplex	Telegraph	Through MTD-1
UZO-4800	YeS-8135	USSR	50, 100, 200, 2,400, 4,800	KOI-8	Cyclical with Polynomial $x^{16}+x^{12}+x^5+1$	Duplex	Assigned Telephone	Through Modem-2400
UZO-48000	Yes_8140	USSR	24,000, 48,000	ITC-5 or KOI-8	Cyclical with Polynomial $x^{24}+x^{23}+x^7+x^2+1$	Duplex	Broad-Band	Through Modem-48000

interlink with the computer input-output channel which performs parallel data exchange between the data transmission multiplexor and the computer; a two-channel switch that makes it possible to work with two multiplex channels and different models of YeS computers; and, line adapters that take account of the special characteristics of the user stations that are being connected. The number and composition of line adapters depends on the type of data transmission multiplexor and the configuration of the remote data processing system.

Table 12.3 below gives the technical characteristics of data transmission multiplexors that have been realized within the YeS remote data transmission system. All the multiplexors are connected to the multiplex channel of the computer through a standard input-output interface; they are connected to the data transmission equipment through an S2 interface and directly to telephone channels through an S1 interface.

The MPD-1A (YeS-8400) data transmission multiplexor allows up to 15 different user stations to be connected to the computer through communications channels. These stations may be AP-61, AP-63, AP-70, RTA, and any model of YeS computer which includes an MPD-1A. The transmission speed by physical lines and telegraph communications channels is 75 bauds; by telephone channels it is 200-4,800 bauds.

Telegraph communications channels and physical lines are connected to the multiplexor either separately or through a signal conversion unit; telephone channels are connected through a Modem-200 or Modem-2400. All incoming and outgoing connections are made automatically. The only exception is establishing outgoing connections on switchable telegraph lines. This operation is done, on the initiative of the computer, by the operator, who manually dials the number outputted by the computer.

The following line adapters have been developed for the MPD-1A: AD-1 for work with page-printing telegraphs through switchable telegraph communications channels; the AD-2 for work with the AP-70 through assigned telegraph channels; the AD-3 for work with AP-61 and AP-63 through the Modem-2400 and assigned telephone channels; the AD-44 work with AP-1 and AP-70 through telephone channels and the Modem-200. These adapters provide a semiduplex mode of data exchange. The AD-5 adapter carries on communications among models of YeS computers in the duplex mode. The synchronous adapter AD-6 is used for work with all synchronous user stations (AP's) on non-switchable telegraph lines. For the purpose of connecting the line adapters the MPD-1A has an adapter connection block (BPA) that communicates with the block for interlinking with the channel, and an adapter synchronization block (BSkhA) for control of the adapters.

Figure 12.3 below shows one possible configuration using the MPD-1A.

The MPD-3 (YeS-8403) data transmission multiplexor carries on communication between YeS-1020, YeS-1030, YeS-1050, and YeS-1060 computers with four user stations. The AP-1, AP-3, AP-11, AP-70, page-printing telegraph, or YeS computer with MP-3 may be used as user stations.

Table 12.3. Technical Characteristics of Data Transmission Multiplexors

Model	Code	Manufacturing Country	Number of Connected User Stations	Type of User Station	Transmission Speed, bauds	Method of Protection Against Errors
MPD-1A	YeS-8400	USSR	15	AP-1; AP-61; AP-62; AP-63; AP-64; AP-70, and Others; MPD-1A, telegraph	75, 4,800	Matrix and Cyclical Codes
MPD-2	YeS-8402	USSR	8-176	AP-1; AP-2; AP-4; AP-14; AP-31; AP-50; AP-61; AP-63, and Others; MPD-2, telegraph	50, 4,800	Linear-Transverse "Check" and Cyclical Code
MPD-3	YeS-8403	USSR	4	AP-1; AP-2; AP-3; AP-4; AP-70; MPD-3, and Others, telegraph	50, 4,800	" "
MPD-1	YeS-8401	Bulgaria	32-64	AP-1; AP-31; AP-61; AP-62; AP-70; MPD-1, and Others, telegraph	50, 2,400	-
MPD-4	YeS-8404	East Germany	12	AP-1; AP-5; AP-6; AP-62; AP-70	20-1,200	Matrix Code
UMPD	YeS-8421	Hungary	20	AP-1; AP-62; AP-64, and Others; MPD, telegraph	50-200	-

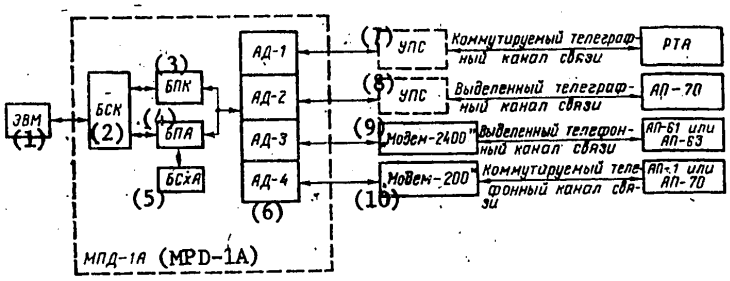


Figure 12.3. Configuration of the YeS Remote Data Transmission System Using the MPD-1A.

Key: (1) Computer;
(2) Block for Interconnection with the Input-Output Channel of the Computer;
(3) BSK;
(4) Block for Connection of Adapters (BPA);
(5) Block for Synchronization of Adapters (BSkhA);
(6) Line Adapters Nos 1-4;
(7) Signal Conversion Device ← Switchable Telegraph Communications Channel → Page-Printing Telegraph;
(8) Signal Conversion Unit ← Assigned Telegraph Communications Channel → AP-70;
(9) Modem-2400 ← Assigned Telephone Communications Channel → AP-61 or AP-63;
(10) Modem-200 ← Switchable Telephone Communications Channel → AP-1 or AP-70.

This multiplexor has the following types of line adapters:

a. the TA-1 start-stop adapter, with which AP-1 and AP-70 user stations are connected to the MPD-3 through the Modem-200 and AVU-TF automatic call device on switchable and assigned telephone channels;

b. the TA-2 start-stop adapter, which facilitates communication with user stations that have telegraph equipment by the MPD-3 on assigned telegraph channels. Only hardware control of the information being transmitted is carried out in this case;

c. the SA-1 synchronous adapter, through which the AP-11 or the MPD-3, which is similar to it, are connected to the multiplexor on assigned telephone channels. Communication passes through the Modem-2400. The method of protection against errors is use of a cyclical code with a forming polynomial $x^{16} + x^{12} + x^5 + 1$;

d. the AA-1 asynchronous adapter, by which AP-2 and AP-3 user stations are connected to the given multiplexor through the UZO-1200 error protection device and the Modem-1200 on switchable and assigned telephone channels.

The MPD-3 can be provided with various types of line adapters to provide a semiduplex mode of data exchange. For work in the duplex mode the DA-1 line adapter, which operates with user stations not included in the set of YeS equipment, and DA-2 line adapters that provide intermachine exchange are used.

Information is transmitted on communications channels at speeds between 50 and 4,800 bauds. Figure 12.4 below shows the configuration of a remote data processing system using the MPD-3.

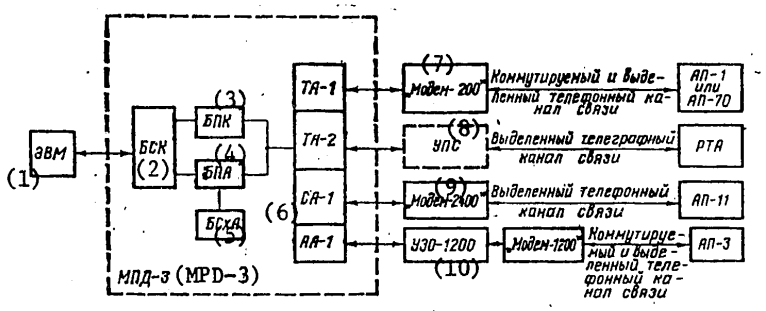


Figure 12.4. Configuration of the YeS Remote Data Processing System Using the MPD-3.

Key: (1) Computer;
(2) Block for Interconnection with the Input-Output Channel of the Computer;
(3) BSK;
(4) Block for Connection of Adapters (BPA);
(5) Line Adapters Nos 1-4;
(6) SA-1 Synchronous Adapter;
(7) Modem-200 ← Switchable and Assigned Telephone Communications Channel → AP-1 or AP-70;
(8) Signal Conversion Unit ← Assigned Telegraph Communications Channel → Page-Printing Telegraph;
(9) Modem-2400 ← Assigned Telephone Communications Channel → AP-11;
(10) UZO-1200 ↔ Modem-1200 ← Switchable and Assigned Telephone Communications Channel → AP-3.

The MPD-1 (YeS-8404) data transmission multiplexor provides connections with user stations on 32 (with possible augmentation to 64) switchable and assigned telephone channels and non-switchable telegraph channels, as well as on physical lines. The speed of transmission may be 50, 75, 100, 200, 600, 1,200, and 2,400 bauds.

The MPD-1 has three types of adapters. The TA-1 works with AP-1 and AP-70 stations through switchable and non-switchable telephone channels. The TA-2 works with telegraphs, and the TA-3 operates with AP-61 and AP-63 stations through non-switchable telephone communications channels. The MPD-1 includes a memory block that has two modules with 32 addressable cells apiece, containing the control words of the lines. One module has a capacity of 2,112 bits and is linked to the TA-1 and TA-2 adapters; the other, with the capacity of 2,240 bits, is connected to the TA-3.

The MPD-2 (YeS-8402) programmable data transmission multiplexor is designed to organize large systems within the Unified System of Computers. It carries on data exchange between computers and all types of data transmission multiplexors and user stations included in the hardware of the YeS remote data transmission system.

The MPD-2 makes it possible to connect between eight and 176 communications lines, increasing by increments of eight. Non-switchable and assigned telephone communications channels and physical lines are used for this period. The transmission speeds are 50, 100, 200, 600, 1,200, 2,400, and 4,800 bauds.

Unlike the hardware-type data transmission multiplexors we have considered, in which the assortment of user stations is always limited, the MPD-2 can connect various types of user stations. This advantage greatly increases the flexibility of the system, helps broaden it when necessary, and increases the number of connectable user stations.

New user stations are connected by copying a program into the memory of the multiplexor without making changes in its circuitry. The MPD-2 includes a disk memory unit with a capacity of 4,096 72-bit words and a microprogram control block. The multiplexor uses these devices to perform functions related to the execution of concrete exchange algorithms. In addition to these devices the MPD-2 has a main memory unit which stores the control and information words of the lines for all communications channels.

The UMPD remote multiplexor (YeS-8421) concentrates 20 non-switchable telegraph communications channels working at a speed of 50 bauds and having user stations at the ends into one non-switchable telephone line with a speed of 1,200 bauds. The use of this multiplexor makes it possible to significantly reduce the number of communications channels and greatly increase the efficiency of their use. One of the possible configurations of a remote data processing system using the UMPD is shown in Figure 12.5 below.

Because the functions of the remote data processing system related to analyzing titles, establishing orders, editing messages, and polling the state of the user stations are done by computer, further development of the YeS remote data transmission system presupposes development and introduction of special communications processors and, in connection with this, the redistribution of remote data transmission functions.

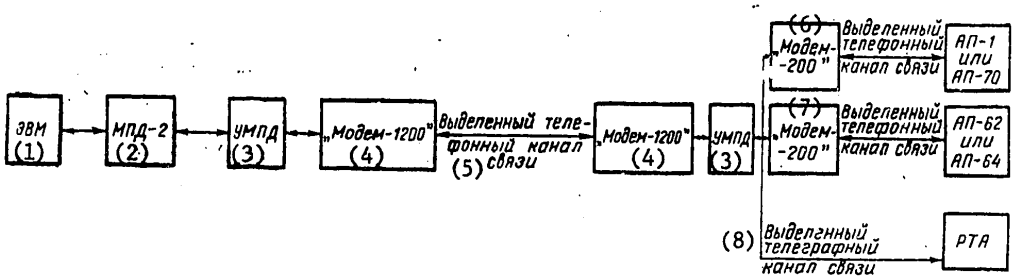


Figure 12.5. Configuration of a YeS Remote Data Processing System Using the UMPD.

Key: (1) Computer;
(2) MPD-2;
(3) UMPD;
(4) Modem-1200;
(5) Assigned Telephone Communications Channel;
(6) Modem-200 ← Assigned Telephone Communications Channel → AP-1 or AP-70;
(7) Modem-200 ← Assigned Telephone Communications Channel → AP-62 or AP-64;
(8) Assigned Telegraph Communications Channel → Page-Printing Telegraph

---

The use of communications processors in place of data transmission multiplexors helps reduce the workload of the computers and increase system flexibility and reliability and the possibility of employing them as message switching centers in computer networks. In most cases communications processors are minicomputers with special software. The functions of interlinking the minicomputers with the central computer and transmission units are performed by linear adapters.

[116-11,176]

11,176
CSO: 1863

PUBLICATIONS

UDC 681.3

NEW BOOK ON AUTOMATING CONTROL OF PRODUCTION PROCESSES

Moscow OSNOVY AVTOMATIZATSII UPRAVLENIYA PROIZVODSTVENNYMI PROTSESSAMI in Russian 1980 (signed to press 15 Oct 80) pp 2, 4-5, 358-359

[Annotation, excerpt of preface, and table of contents of book "Fundamentals of Automating Control of Production Processes" by Vinfrid Kal'fa, Valeriy Valentinovich Ovchinnikov, Oleg Mikhaylovich Ryakin, Gans-Yurgen Sebastian, and Vladimir Vasil'yevich Smirnov, Izdatel'stvo "Sovetskoye radio", 8,000 copies, 360 pages]

[Excerpts] Annotation

This book considers automated data processing systems that provide optimal control of production processes on the basis of mathematical models that take into account the indeterminacy of a number of parameters of the control process. The development of data processing systems in two directions is shown: the organization of large collective-use systems, and the establishment of systems of small computers. The authors describe means and methods of improving the reliability of information in different stages of its processing.

The authors present results obtained by them in the development and introduction of means for automating control at enterprises in the Soviet Union and East Gemany.

The book is intended for scientific workers and engineers who are specializing the the fields of control system design and data processing.

There are 79 illustrations, 42 tables, and 134 bibliographic entries.

Preface

In the first section of the book, Chapter 1 considers the basic tasks of designing automated control systems for production processes, and then Chapter 2 describes certain new mathematical models of control of processes characterized by a large volume of manual labor and lack of intermediate data on the performance of particular operations (movement of freight, constructaion-installaion work, and other jobs). Techniques

are developed for this kind of model to optimize control processes with due regard for incomplete information on the controlled objects. These techniques have been used at a number of construction industry enterprises in East Germany and offer new possibilities of raising the productivity of these processes.

The second part of the book analyzes the principles of constructing a collective-use data processing system capable of serving a number of remote users at the same time and oriented to models of the YeS [Unified System] family of computers (Chapter 3). Such a system based on a computer center has been developed and introduced in East Germany. Chapter 4 considers the questions of organizing and using data banks in integrated data processing systems for control of production processes (using examples from enterprises in the USSR, East Germany, and West Germany).

The third part of the book is devoted to the use of "small" third-generation computer equipment to automate control. Chapter 5 considers the questions of using microcomputers and microprocessors, while Chapter 6 deals with the special characteristics of using minicomputers. Substantial attention is devoted to the principles of modularity, mainlining, and microprogramability for this class of computers and to the organization of input-output interfaces for "small" computers.

The first part (Chapters 7-9) considers the primary methods of insuring data reliability in automated processing systems. This is accomplished by software and hardware means accompanying the computing processes. Chapter 7 describes methods of noise-immune input of initial data. Chapter 8 deals with the basic methods used to insure the reliability of storage of data files. Chapter 9 covers methods of monitoring computations in the central processor.

The preface and Chapters 1 and 5 were written by V. V. Ovchinnikov (USSR). Chapter 2 is written by G.-Yu. Sebastian (East Germany). Chapter 3 was written by V. Kal'fa (East Germany). Chapters 4 and 6 were written by V. V. Smirnov (USSR). O. M. Ryakin (USSR) wrote Chapters 7-9. In addition, Section 3.1 was written jointly by V. V. Ovchinnikov, G.-Yu. Sebastian, and V. Kal'fa.

The authors are grateful to their reviewers, corresponding member of the Kazakh SSR Academy of Sciences V. M. Amerbayev, doctor of technical sciences and professor V. A. Gorbatov (USSR), and doctor of natural sciences and professor A. Sheffer (East Germany). The authors tried to make maximum use of their suggestions and remarks in writing this book.

Table of Contents

11,176
CSO: 1863

UDC 681.3

EQUIPMENT OF COLLECTIVE-USE DATA PROCESSING SYSTEM REVIEWED

Moscow OSNOVY AVTOMATIZATSII UPRAVLENIYA PROIZVODSTENNYMI PROTSESSAMI in Russian 1980 (signed to press 15 Oct 80) pp 89, 99-102, 107-112, 118-119

[Excerpts of Chapter 3 of book "Fundamentals of Automating Control of Production Processes' by Vinfrid Kal'fa, Valeriy Valentinovich Ovchinnikov, Oleg Mikhaylovich Ryakin, Gans-Yurgen Sebastian, and Vladimir Vasil'yevich Smirnov, Izdatel'stvo "Sovetskoye radio", 8,000 copies, 360 pages]

[Excerpts] Chapter 3. Collective-Use Data Processing Systems

3.2. Structure of the Collective-Use Data Processing System (CUDPS)

General Diagram of the CUDPS

Figure 3.7 below shows the general diagram of the CUDPS. The processes are linked to the system through terminals; the terminals may be used by just one process or by different processes sequentially. But for each process there will be at least one dialog module (or one problem program) containing the algorithm for control of data input-output. This diaglog module should have the capability of referring to the data bank and/or data array affiliated with one or several processes.

The operating system has two levels of data control: the problem program — data arrays/data bank, and the problem program — terminal. Both levels are shown in Figure 3.7 where T_1, ..., T_m are the terminals; F_1, ..., F_n are the common arrays of data for the problem programs and system programs of the operating system; UF_1, ..., UF_m are user files; $\Pi\Pi_1$, ..., $\Pi\Pi_R$ are problem programs; and, Π_1 ..., Π_J are the processes. General control of the system is exercised by the operating system (shown by double arrows).

Thus, the operating system sees that each process can use the CUDPS as its own computing machine with virtual terminal addressing. To expand the capabilities of the CUDPS it is necessary to add the operating system and system software. A change in the functional capabilities of the CUPDS is achieved by reorganizing the structure of the operating system without redesigning the hardware. Many books and articles have been written on questions of the work of different types of operating systems. The best, from the standpoint of the CUDPS, is monograph [7].

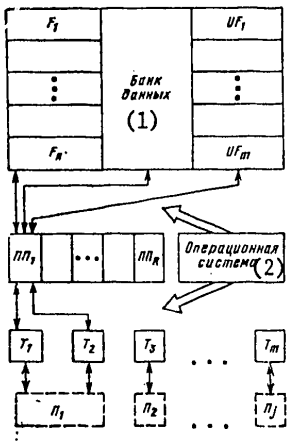


Figure 3.7.

Key: (1) Data Bank;
(2) Operating System.

We will note here only the most important functions of an operating system for a CUDPS, namely: interrupt control, organization of exchange with external data storage, analysis of errors and emergency situations in the computation process, and rapid retrieval of system control and processing programs from the library and putting them in a ready state. The problems of insuring protection of the operating system against destruction when an unprepared user is in dialog with the CUDPS occupy a special place.

The main thing necessary to execute the problem programs is increasing the working speed of the central processors. The YeS-2040 processor, for example, multiplies numbers with floating decimal points in 10 microseconds with a memory volume of 1,024 kilobytes. In terms of capabilities this model is at the lower threshold for use as a central processor in a CUDPS.

Processors

Processors in the CUDPS perform the following functions:

a. processing problem and system programs;
b. control of data transmission channels in the "terminals-concentrator" contour;
c. control of channels for transmission between main and external memory (channel control);
d. control of interrupts based on multiprogramming;
e. storage of data in main memory.

An example of a multipurpose processor that transmits data by telephone at a speed of up to 48 kilobauds is the MPD-4 (YeS-8404) minicomputer produced by the Robotron Association in East Germany.

Terminals for work on a real time scale are used in control systems for complex production processes that envision the possibility of automatic

collection of data on situations in production processing. Data are transmitted to the CUDPS, processed, and the results (parameters controlled) transmitted back to the terminals. The raw data may be transmitted from sensing devices, automatic monitoring and measuring devices, or operators. When working with such terminals it is essential to provide special linkages between the measurement complexes or devices and the hardware of the base computer.

Minicomputers and microcomputers with microprogramming can be used to increase the speed of data transmission equipmant. The YeS-8505 user station for collection of data in different sections of production processes, shown in Figure 3.12 below is an example. These terminals are used in systems to

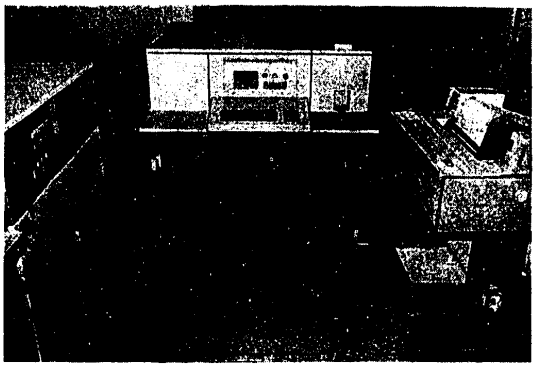


Figure 3.12

---

monitor and control transportation, the condition of the environment, the supply of water, electricity, and gas, care of patients, and so on. Table 3.6 below gives more detailed characteristics of the YeS-8505 user station.

Table 3.6. Characteristics of the YeS-8505 User Station.

Total Number of Dispatcher Points (DEP) for Recording Data on Ongoing Production Situation	15
Method of Recording	
DEP-A	Digital Input-Output
DEP-B	Digital Input, Alphanumeric Output
DEP-C	Alphanumeric Input-Output

[Table continued next page]

[Table 3.6 continued)

Method of Data Transmission (Through YeS-8002, 8006 or UPSN, and YeS-8028 Modems)	Conversion of Sequential Code To Parallel Code in Conformity with Standard CCITTV24
Method of Linkage To MPD-4 Data Transmission Multiplexor (YeS-8404)	Through a 12ESEP Device by "Start-Stop" Signals or Through a Standard SIF1000 Interface
Directivity of Interface	Semiduplex
Speed of Data Transmission (for 12ESEP)	200, 600, 1,200 Bits/Second
Transmitted Code	KOI-7
Error Protection	Standard VRS/LRS Code
Dimensions of:	
DZA Decentralized Polling Unit	800×500×1,130 mm^3
Data Array Generator	880×500×960 mm^3
DEP Dispatcher Point	1,210×805×1,065 mm^3
AST Asynchronous Control Device	540×380×290 mm^3
Length of Connections	
Direct Connection DEP-User Station	20 Meters
Linkage of DEP-Expander and DUE-User Station	1,000 Meters

Independent automatic systems for monitoring, measurement, and control which do not have remote communications with the base computer have become rivals of these terminals (taking into account the reduced expenditure for electronic elements and memory units). The automatic systems have an advantage if direct control of the object through the CUDPS is not required and the data may be delivered to the computer center on magnetic tape. This could apply, for example, to servicing machines and mechanisms working under field conditions.

Specialized terminals for production operations depend on the software of the controlled process and have numerous modifications. The advantages of their application in the CUDPS are that data is collected directly from the source, time losses for collecting information are reduced, and precision is greater. The specialized unit that monitors commercial transactions is an example. It can include a reading unit, a keyboard with numbers and letters, a printer, a display for operational information on transactions, buffer memory, cassette accumulators, a simple arithmetic-logical unit based on a microprocessor, programmable logical matrices, a synchronization block, and a block of modems and interlinking devices. The last four elements may be included in minicomputers or microcomputers.

Data is collected by means of a decimal digital keyboard (10-20 bit positions), special working keys, a table reading device, a monitoring light indicator, and logical and control memory units. These terminals are used

in banks, systems for reserving places in libraries, hospitals, and polyclinics, in transportation, and elsewhere.

"Intellectual" terminals are devices that contain minicomputers, microcomputers, or microprocessors and provide for programming the input-output of complex assignments. The contemporary "intellectual" terminal may have its own main memory (with the possibility of enlarging it by supplementary external storage) and a processor. The terminal is connected to the base computer and its external memory through appropriate interlinking devices. It is capable of performing control programs for transmission of compacted data independently. As a result, time expenditures for data transmission and loading communications channels are significantly reduced.

Figure 3.13 below shows the hardware of the KMU-400, which is part of an "intellectual" terminal of the Baukombunat system for control of production processes designed for an East German construction-installation enterprise that does jobs involving shipping large amounts of large-dimension freight over long distances [10]. The basic units of these terminals and their indicators are shown in Table 3.7 below.

Figure 3.13

Table 3.7. Units of the "Intellectual" Terminal

Main Memory	16•1,024 Words, each of which contain 16 bits; access time 1.3 microseconds
Arithmetic-Logical Unit with Fixed Decimal Point	Performance Time for Operation of Addition — 65 Microseconds

[Table continued next page]

[Table 3.7 continued]

Concentrator	Channel for Connecting 12 Devices To Control Data Input-Output
Channel for Direct Access to Memory	Group Copying of Data Between Memory and Input-Output Devices at a Speed of Up To 1 Million Words a Second
External Accumulator	
Magnetic Drum	98,304 Words
Magnetic Tape	Length — 360 Meters; Width — Roughly 12.6 Millimeters
Magnetic Disk	10-15 Megabytes
Peripheral Devices	
Character Synthesizer	100 Characters per Second
Punched Tape Block	Up to 1,000 Characters a Second
Typewriter	10 Characters a Second
Magnetic Tape Cassette	45,000 Words
Display	Videoton 340
Punched Card Reader	500 Cards a Minute

Data Transmission Devices

An important part of the CUDPS is devices to transmit data between terminals and the base computer or among several computers. The volume of data to be transmitted and the time necessary for the response depends significantly on the software of the object (process). When the base computer is significantly removed from the object of control the process of data transmission requires solving a number of fairly complex problems:

1. the use of telephone channels to transmit discrete information;
2. coordination of the carrying capacities of telephone and telegraph channels with the carrying capacity of the computer;
3. reducing the level of errors occurring in channels to acceptable values.

The difference between the carrying capacities of communications channels and computers (see Figure 3.14 [not reproduced]) is compensated for by buffer memories which are used for both transmitting and receiving messages. The volume of buffer memory differs. It depends on the method of transmitting symbols from terminals to the computer and in the other direction. With asynchronous transmission from a teletype it is sufficient to store just one symbol; for synchronous transmissions from a display 1,024 symbols must be stored. In the latter case the channel is free to transmit data from other terminals.

The package of messages is first accumulated in buffer memory, and then transmitted to the channel. This job is often done by data transmission multiplexors (MDP's) in the base computer [11].

Modulator-demodulators are used to coordinate the signals with the frequency band of the communications channels of the system. The telephone network usually has a frequency range from 300 to 3,400 Hertz. Therefore, it is necessary to convert signals to high frequency signals by means of amplitude, frequency, or phase modulation. The high frequency signal is demodulated at the user station and converted to digital code. Because two-way communication between the base computer and peripheral devices is usually required, both devices (the modulator and demodulator) are joined in a single design unit called the modem. Without a modem transmission is possible only for short distances through cable or wire lines. Data on modems and transmission devices by cable lines are given in Table 3.8 below [12].

Table 3.8

Type of Device	Speed, bauds	Work Regime	Method of Data Transmission	Number of Lines
YeS-8002	200	Duplex	Asynchronous	2
YeS-8006	1,200	Semiduplex	Synchronous	2×4
YeS-8028	2,400	Semiduplex	Synchronous	2×4
	9,600	Duplex	Asynchronous	

External Memory Units

Table 3.11

Computer Model	Firm	Type of Accumulator	Average Access Time To Accumulator, microseconds	Exchange Speed, kbit/sec	Recording Density, bit/mm	Maximum Capacity of Accumulator M Bytes
YeS-5052	Elka (Bulgaria)	Disk	90	156	35	7.25
YeS-5061	Elka (Bulgaria)	Disk	90	312	88	29
3470	Siemens (FRG)	Disk	20	-	-	420
YeS-5017	Zeis (GDR)	Tape	5	64	32	732 (15 mm gap in block)

[111-11,176]

11,176
CSO: 1863

UDC 681.3

CONTROL SYSTEMS FOR DATA BANKS AND BASES REVIEWED

Moscow OSNOVY AVTOMATIZATSII UPRAVLENIYA PROIZVODSTVENNYMI PROTSESSAMI in Russian 1980 (signed to press 15 Oct 80) pp 142, 162, 164-171

[Excerpts of Chapter 4 of book "Fundamentals of Automating Control of Production Processes" by Vinfrid Kal'fa, Valeriy Valentinovich Ovchinnikov, Oleg Mikhaylovich Ryakin, Gans-Yurgen Sebastian, and Vladimir Vasil'yevich Smirnov, Izdatel'stvo "Sovetskoye radio", 8,000 copies, 360 pages]

[Excerpts] Chapter 4. Data Banks in Control Systems for Production

Table 4.1.

Type of Data Bank Control System	Developer-Country	First Year of Use	Minimum Capacity of Main Memory, K bytes	Mode of Use	Type of Computer
GIS	USA	1966	128	Batch	IBM/360-40, 50
MIS/360	USA	1967	256	Batch, time-sharing	IBM/360-40, 50
DBOMP	USA	1967	32	Batch	IBM/360-25
MARK IV	USA	1968	384	Batch	IBM/360-40, 50
MARS III	USA	1968	512	Batch	CDC 3170, 3300, 3500
MARS IV	USA	1969	256	Batch, dialog	CDC 6400, 6500, 6600
DMS 1100	USA	1969	384	Batch, dialog	CDC 6400, 6500, 6600
SESAM SIEMENS	FRG	1969	128	Batch, dialog	IBM/360-40 4004/135
POLIS	FRG	1970	64	Batch, dialog	IBM/360-40
CICS	USA	1970	128	Batch, dialog	IBM/360-40
FMS-8	USA	1971	384	Batch, dialog	UNIVAC 1000
IMS/2	USA	1971	128	Batch, dialog	IBM/360-40 50
DIS/DISDIA	FRG	1971	128	Batch, dialog	IBM/360-40
ADABAS	FRG	1971	128	Batch, dialog	SIEMENS 4004, IBM-360/40
BASTEI	GDR	1972	128	Batch	YeS-1040

[Table continued next page]

[Table 4.1 continued]

Type of Data Bank Control System	Developer-Country	First Year of Use*	Minimum Capacity of Main Memory, K bytes	Mode of Use	Type of Computer
BANK 2000	USA	1973	256	Batch, dialog	-
UNA 1071	FRG	1973	256	Batch, dialog	-
SINBAD	USSR	1975	128	Batch	YeS Computers
NABOB	USSR	1975	128	Batch	YeS Computers
BAZIS	USSR	1975	256	Batch, dialog	M-4030
BANK	USSR	1976	64	Batch	YeS Computers
SIOD 1, 2	USSR	1976	32	Batch	YeS Computers
OKA	USSR	1976	138	Batch, dialog	YeS Computers
	USSR	1976	256	Batch, dialog	YeS Computers
INES	USSR	1979	512	Batch, dialog	YeS Computers
DIAMS	USSR	1977	16	Batch	SM-3, SM-4
IMAGE	USA	1976	128	Batch, dialog	HP-3000

* Determined by latest publications

The BANK data base control system is a set of languages and software for organizing data banks of random composition on magnetic disks for the YeS computers. The program package of the system is designed for work under the control of the YeS disk operating system. The BANK system presupposes two levels of use:

1. the administrator level, in which role it operates as a programmer who creates and maintains the data base;

2. the problem programmer level; this is the level at which the users of the data bank who form requests for processing and shaping output reports work.

The languages of the BANK system consist of the language for describing the structure of the data base and the language for access to store data. Both languages are sets of macrocommand. The problem programmer can write his own requests using the macrocommands of the access language; in this case the level of access corresponds to the level of the logical input-output system of the YeS disk operating system.

The logical level of data of the BANK system consist of fields, entries, and chains. There is a logical address corresponding to each logical entry which determines its relative position in the data base. A chain consists of several types of entries. Entries of one type (the maximum number of types is 255) should have identical composition and a fixed length. Within a chain entries are interconnected by logical addresses. The chains may contain entries which are physically located on the disks of different packages. The same entry may be included in several chains. In each chain one

entry is always the main one: the chain begins and ends with it. Within the chain entries may be arranged by order of increasing or decreasing values of the entry fields declared by key entries. The connections that form the chain may be one-way or two-way and may also provide a reference to the main entry of the chain. Thus, data structures of the list, tree, and network types are possible.

In physical terms, the data base controlled by the Bank system may consist of several files and take up several packages of disks. The whole memory of the data base is broken into blocks which are called pages. The maximum number of blocks of the data base should not exceed 65,535. The size of a block is limited by the length of the disk track. Each file should consist of blocks of the same size. The number of different types of entries within a block is arbitrary. Each block contains a mandatory entry formed by the system which indicates the main entries of the list of free sectors of the block (tracks) and lists of entries of the given block. The system uses these lists in record-keeping and distributing memory for new entries.

The administrator of the base selects the method of locating entries in blocks of the base from the set of methods offered by the BANK data base control system:

a. for locating an entry the programmer should indicate the correct address, the logical address: the block number and address of the entry in the block;

b. use of the system randomization procedure , which computes the number of the block by the value of the entry field declared by the key;

c. entries are arranged in the same block where the main entry of the chain is located or close to it;

d. filling blocks in order beginning with the first;

e. filling blocks, skipping intervals assigned by the user.

In the data description the developer should also assign the technique for retrieving entries: direct, using the key randomization procedure, or through the chain.

The problem programmer when developing the program must envision memory fields in it for interaction with the BANK data base control system:

a. at least two buffers whose dimensions are equal to the largest block being used;

b. working fields for all entry fields and keys with which he will work;

c. four communications fields (four bytes long each): a field containing the logical address of the last entry used;

a field to notify the programmer of error situations;
two more fields that assign the beginning and final addresses of the block (or entries) for group retrieval.

Access to the BANK data base control system is possible from programs written in Assembler (using declarative macrocommands), Cobol, and PL/1 (using the "Call" statement) by indicating the addresses of the tables of parameters. The BANK data base control system provides the programmer with the following procedures: "Open," "Close," "Store," "Find," "Remove," "Modify," "Copy," and "Transfer on Condition."

The description of the data base may be done by seven macrocommands with key parameters. The order of description macrocommands is fixed.

Protection of the data base involves monitoring the parameters communicated by the user when referring to the base. A system log and restoration program are used to restore integrity. The system log is put on magnetic tape and all changes over a certain period of time are recorded on it. The creation of copies of the data base should be envisioned by organizational measures.

The BANK data base processing system contains programs to form and correct data bases, programs for reorganization and restoration, and programs to compile tables and describe the structure of a data base.

Three modules are used for the work of the problem program: the base description module, the procedural module (in one of two variations: for retrieval only or for retrieval and updating), and the data base control system resident module. The resident module occupies four kilobytes; the procedural model occupies a maximum of 14 kilobytes of main memory.

The BANK data base control system was used for setting up a specialized bank of normative-reference data at an industrial enterprise [8].

Examples of Data Banks

The Data Bank Used at Machine Building Enterprises

An interesting example of this type of data bank is the BASTEI system [1, 27] developed by the scientific research center of the East German Robotron Combine.

The raw data for the BASTEI bank is design specifications and production route cards. These documents are the basis for organizing the production process at machine building enterprises.

The developers selected key specifications from the broad range of different types of specifications because, as experience has demonstrated, the volume of essential data here is relatively small. Moreover, the use of them allows changes to be made in the data with comparative simplicity.

In addition to the raw data mentioned above, there must also be data on work positions and all materials, parts, assemblies, and articles (called objects) used and manufactured at the combine.

Each specification contains data relating to the parts and assemblies of one level of assembly; only data that reflects the structure of the assembly on this level (list of parts and their number) is given. All the remaining information characterizing the parts of the assembly or the initial materials is omitted from the particular specifications. This same principle is applied to all other types of data: objects, industrial routing cards, and work positions.

The industrial routing card corresponds to assembly specifications and has data on the order of operations performed during the manufacture of an object made up of parts or assemblies on one level of assembly. This may include the number of the work position, preparation and conclusion time, the type of norm, the job rating, and instructions on performance of operations. Data depending on the work position at which the operation is performed is not included in the industrial routing card.

Data on work positions includes information on each work position, the production capacities of the equipment, calendar plan norms, and the like.

Data on objects, which means all objects handled in the production process, occupies a special place. This comprises raw material, semifinished materials, purchased parts, standardized parts, assemblies, and articles, and the like.

All the data used by the BASTEI system is recorded in direct-access memory (replaceable disks) in four files: objects, specifications, industrial routing cards, and work positions. Each installation is assigned a code in advance. The connections among files are rigid and give the addresses of the corresponding entries in the given or other files. Chain addressing is widely used. Its basic purpose is to attempt to minimize large, labor-intensive sorting jobs which are inevitable with conventional file data organization.

The BASTEI system has a complex of control programs that are used for loading, correcting, and reorganizing data. Users are offered a set of macro-commands for performance of practical tasks.

The record of each file consists of two parts: information and control. The information part contains the data considered above and is called data for the users; the control part contains the addresses of the linkages that are used by the control programs. The control parts of each entry are formed in the process of loading raw data.

The raw data for the system is prepared on punched cards and loaded into memory in a definite order because of the need to know certain information about files already loaded when loading the next file.

Data loading begins with feeding data on the object and construction of the file of objects in a predetermined area on replaceable magnetic disks. A distinct file entry corresponds to each object. The control part of the entry reserves a place for the addresses of the first application of the particular object in the specification file and the industrial routing card file.

After the subject file is loaded data on specifications is fed and the specifications file is created. Each part has its own entry. The control part of each entry of this file forms the address of the corresponding entry in the object file, the address of the next part of this assembly or article, and the address of the entry that determines the next use of the particular part. Thus, a connection is established with the entries of the object file and a chain list of all assemblies or articles which include the given part or assembly is formed. At the same time a whole number corresponding to the degree of inclusion of this object is formed for each object (part, assembly). The degree of inclusion for articles is taken as equal to zero, while for the parts and assemblies of each previous level of assembly it increases to one. The degree of inclusion is used for monitoring during the process of loading and correcting the specifications file and for controlling the beginning of production of articles during work with the system.

The file of work positions is loaded next. After data on each work position is fed, the control part of each entry reserves fields for the address of the first entry of the industrial operation and the address of the first entry for use of the work position in the industrial routing card file. These fields will be filled during the loading of this file, and thus a connection will be established between the work position file and the industrial routing card file.

The industrial routing card file is the last to be loaded because a knowledge of the position of entries in the object and work position files is necessary to shape its entries. For each part manufactured at the combine there is a corresponding list of industrial operations, work positions, and other information. The control part of each entry contains the address of the corresponding entry in the object file, the address of the entry in the file of work positions for each operation, the address of the entry of the next use of the given work position, and the address of the next operation in the sequence of industrial operations for the particular part.

The BASTEI system enables users to receive specifications and industrial routing cards in different forms of notation, cards for the applicability of parts, and cards for use of work positions. Processing the specifications file makes it possible to receive three types of specifications: assembly, structural, and composite. These specifications are essential and convenient documents for organizing and controlling production, for design bureaus, design changes, for material-technical supply subdivisions, and for calculating requirements of parts and materials.

The cards for applicability of parts are also produced by processing the specifications file. The following variations of applicability cards are

possible: single-stage, structural, and composite. The structural and composite cards of the applicability of parts indicate the degree of inclusion, which makes these documents graphic and convenient for the technical service concerned with changes in the design of articles and preparation for production.

The industrial routing cards, supplemented with time norms, batch size, and certain other information, may be used as production assignments.

Combined use of the work position file and the industrial routing card file makes it possible to receive a card for the use of the work position.

The organization of data we have considered permits comparatively simple modification and supplementation of stored data. If a certain article must be taken out of production, all that is necessary is to eliminate the corresponding file in the objects file, the industrial routing card file, and the specifications file and correct the addresses of the linkages of part applicability for the particular article in the specifications file and the addresses of linkages in the industrial routing card file. These actions are performed automatically by the BASTEI system.

The BASTEI system was employed in the USSR to develop the automated control system at the First Moscow Clock Plant [23].

[111-11,176]

11,176
CSO: 1863

UDC 681.3

SPECIFICATIONS OF MICROPROCESSORS LISTED

Moscow OSNOVY AVTOMATIZATSII UPRAVLENIYA PROIZVODSTVENNYMI PROTSESSAMI in Russian 1980 (signed to press 15 Oct 80) pp 175, 192-195

[Tables 5.6, 5.7, and 5.8 of book "Fundamentals of Automating Control of Production Processes" by Vinfrid Kal'fa, Valeriy Valentinovich Ovchinnikov, Oleg Mikhaylovich Ryakin, Gans-Yurgen Sebastian, and Vladimir Vasil'yevich Smirnov, Izdatel'stvo "Sovetskoye radio", 8000 copies, 360 pages]

[Excerpts] Chapter 5. Data Processing Systems Based on Microprocessors and Microcomputers

Table 5.6

Type of Microprocessor and Company or Manufacturing Country	L,bits	t_k, μsec	N_K	N_p	V_o, K words	V_y, K words	P_{use}, mwt
EA 9002 Electronic Arraus	8	0.5	60	26	4	8	700
F-8 Fairchild Semiconductor	8	1	70	65	64	64	500
CP-1600 General Instru.	16	0.8	87	8	64	64	750
8080 Intel	8	0.5	78	7	64	64	780
8080 Texas Instruments	8	0.5	78	7	64	64	780
9900 Texas Instruments	16	1.3	69	15	64	64	1,200
6800 Motorola	8	2	72	5	64	64	600
Elektronika 60, USSR	16	5	78	16	4	33	1,200
Elektronika K110, USSR	8	3-5	78	24	64	64	1,800

	N_{np}	N_{int}	N_{dac}	Change	Cross	Comp
EA 9002 Electronic Arraus	1	3	1	yes	yes	yes
F-8 Fairchild Semiconductor	3	4	2	yes	yes	yes
CP-1600 General Instru.	3	3	2	yes	yes	yes
8080 Intel	3	4	2	no	yes	yes
8080 Texas Instruments	2	4	2	yes	yes	yes
9900 Texas Instruments	3	4	3	no	yes	no
6800 Motorola	1	4	2	no	yes	yes
Elektronika 60, USSR	3	3	no	yes	yes	yes
Elektronika K110, USSR	3	4	no	no	no	no

Key: N_{pr} - Power source rating; N_{int} - Number of interrupt levels; N_{dac} - Number of channels with direct access to memory; Change - Possibility of changing content of microprograms; Cross - Assembler cross-system; Comp- Compiler from PL-M and Fortran-IV.

Table 5.7

Type of Microprocessor and Company or Manufacturing Country	Technology	L, bits	t_k, μsec	N_K	N_p	V_o; K words	V_y, K words
PB-96 Digital Laboratories	bipolar	8	0.4	18	18	1	4
3002 Intel	bipolar	2	0.15	40	12	0.5	2
8086 Intel	CMOS	16	0.5	62	16	4	8
IM6100 Inter. Ing.	CMOS	12	1	70	4	0.25	1.5
10800 Motorola	ESL*	4	0.1	128	5	16	64
COSMAC CDP1801/18016 RCA	CMOS	8	1.3	57	17	64	64
SM3000 Scientific Microsystem.	bipolar	8	0.7	8	8	8	16
SX 160 Essex Inter.	I^2L*	4	1.6	41	2	1	2
Elektronika NTs-03, USSR	CMOS, bipolar	16	2-3	192	24	65	128
Elektronika 80NTs, USSR	CMOS	16	6-8	78	8	16	64
Elektronik 8260, GDR	TTL*	16	5.5	112	16	16	64
K580IK80, USSR	n-MOS	8	2.0	120	-	64	64

	P_{use}, mwt	N_{pr}	N_{int}	N_{dac}	Change	Cross	Comp
PB-96 Digital Laboratories	10,000	2	4	2	yes	yes	yes
3002 Intel	1,000	1	8	no	no	yes	yes
8086 Intel	600	2	8	2	yes	yes	yes
IM6100 Inter. Ing.	10	1	8	2	no	yes	yes
10800 Motorola	1,400	2	8	2	yes	no	no
COSMAC CDP1801/18016 RCA	30	1	4	2	no	no	no
SM3000 Scientific Microsystem.	1,500	1	4	3	yes	yes	yes
SX 160 Essex Inter.	120	1	no	no	yes	no	no
Elektronika NTs-03, USSR	180,000	2	8	no	yes	yes	no
Elektronika 80NTs, USSR	9,000	2	8	no	no	no	no
Elektronika 8260, GDR	8,000	2	4	no	no	no	no
K580IK80, USSR	750	2	-	-	- -	-	-

* [Expansion unknown].

Characteristics of Equipment for Communication between Microprocessors and Objects

Table 5.8

Interface Elements	Purpose, Features	Effect of Use
Duplex Amplifiers of the Type HP98032A, Intel 8216, 8212, K580 K55	Normalization of signals in lines of system trunk line	Permits increasing the trunk line with 50 m segments or serving up to 10 terminals (printing, listing programs, graphics)

[Table continued, next page]

[Table 5.8, continued]

Interface Elements	Purpose, Features	Effect of Use
Three- dimensional branches of the type HP98036B with parallel code exchange, Intel 8255	Increase in speed, group camouflage function, distribution of terminals in space	Permits speed of exchange with computer installations of more than 800,000 words/sec for CMOS and bipolar technology with monopolar regime
Three-dimensional branches on semiconductor multiplexor base of type HP98041B, Intel 8253	Same as preceding, but exchange in sequential code	Reduces the cost and improves the reliability of the system by reducing the number of cable links. Speed of data exchange is more than 500,000 words/sec for CMOS and bipolar technology in monopolar regime
Linear amplifiers (analog keys) of the type HP98011A, K590H-1	Coordinating modem with microcomputer	Reduces volume 6-8 times and improves reliability of linked telephone equipment
Linear adapters of the types HP98033A, Z600	Coordination of microcomputer with telegraph channels (purposeful use of optical electronics)	Reduces volume 2-3 times and improves reliability of linked telegraph equipment
Telsat, YeS-8001, 8005, 8010, 8015, and 8019 modems	Coordinating microcomputers with telephone channels, based on digital techniques of analysis and synthesizing harmonic signals	Reduces volume 2-3 times and improves reliability of linked telephone equipment
Discrete communication devices HP98036A, Intel 8251, 8273, K580, IK51	Exchange in sequential code in conformity with GOST-18077-73 in ASU for industrial processes, and the like	Reduces the weight of the cable system and improves the reliability of the ASU for industrial processes, systems to automate experiments, and the like

[111-11,176]

11,176
CSO: 1863

UDC 681.3

MICROPROCESSORS USED IN CONTROL SYSTEMS FOR INDUSTRIAL PROCESSES

Moscow OSNOVY AVTOMATIZATSII UPRAVLENIYA PROIZVODSTEVNNYMI PROTSESSAMI in Russian 1980 (signed to press 15 Oct 80) pp 206-210

[Excerpt of Chapter 5 of book "Fundamentals of Automating Control of Production Processes" by Vinfrid Kal'fa, Valeriy Valentinovich Ovchinnikov, Oleg Mikhaylovich Ryakin, Gans-Yurgen Sebastian, and Vladimir Vasil'yevich Smirnov, Izdatel'stvo "Sovetskoye radio", 8,000 copies, 360 pages]

[Excerpt] 5.4. The Use of Microprocessors and Microcomputers in Control Systems for Production Processes

Control of Industrial Processes

It is common knowledge that industrial processes are constantly becoming more complex. In addition, there are frequent changes in work regimes and swift replacement of equipment and output being produced. Under these conditions the use of conventional means of automation leads to conservatism because this equipment has a very narrow range of parameter control. The use of microprocessors as control organs for local aggregates, in particular to collect and process information and for control, is a fundamental means of changing this situation. Then a change in the nature of the work of some part of an automatic line leads to nothing but replacing one microprocessor program with another. This is especially important when only a part of the industrial process, not the entire thing, is subject to change.

A distinctive feature of the use of microprocessors to control industrial processes is that they can be designed as built-in assemblies of the industrial equipment.

When designing automated lines with microprocessors, however, it is necessary to solve the problem of coordinating all the microprocessors to make them a single control system. In some systems they are concentrated around a more powerful computing machine which guarantees protection and readiness to work in fluctuating conditions and supplies preprocessed and presorted data. In other systems the microprocessors are formed into a homogeneous system with a data bank distributed among them. The second way is usually more complex because a careful time balancing of information flows is necessary.

The most important requirements imposed on microprocessors by industrial processes are high reliability at average speed, small bit format (up to 12 binary characters), and uncomplicated software.

The use of microprocessors in this sphere of production is justified by the following properties of these units: small dimension; low cost; low power consumption, high operating reliability (the time to restore working condition in a system with a microprocessor is much less than for any other kind of system); availability of computing capacities to perform elementary jobs where the use of computers was formerly considered inexpedient (solving simple linear differential equations, integration and differentiation of functions, and the like); and, broad opportunities for specialization owing to the infrequency of change in the structural configuration and the simplicity of replacing programs in permanent memory.

Microprocessors are most often used in external devices and controllers to control switchboards and communications lines with objects, and to devise inexpensive, low-speed equipment (screen consoles and tape-drive mechanisms) and "intellectual" instruments and terminals (automatic testers, registers, and the like). Figure 5.6 below gives a generalized diagram of a microprocessor system for control of an industrial process. The diagram contains two input channels and two output channels linked to the object as well as a computing complex (BK). A distinction is made between the analog input channel (AK), the analog output channel (AB), the digital input channel (ЦК), and the digital output channel (ЦВ). The computing complex has two microprocessors: a 16-bit device to process digital information on control of the object (МПОА) and an eight-bit device to control the input channels and to interlink with the base minicomputer of the upper level of control of the production process (КМП). In addition to the microprocessor the computing complex includes a main memory module with a capacity up to 64 kilobytes and controllers (КУВВ, КУОИ) with external memory devices on magnetic disks and tape (УВП) and data display units (УОИ) (printing, screen, input-output on punched cards and tape).

The analog input channel includes analog sensors ($АД_1$-$АД_i$), amplifier-shapers ($У\Phi_1$-$У\Phi_i$) and analog data transmission multiplexor (АМПА), an analog-digital convertor (АЦП), and a controller for input-output devices (КУВВ). The digital input channel contains the digital sensors ($ЦД_1$-$ЦД_i$), amplifier-shapers, a digital data transmission multiplexor (ЦМПА), and a controller of input-output units.

The analog output channel provides control of automation and remote control devices (actuating mechanisms) at the object (УА и ТМ) by means of a digital-analog convertor. The digital output channel provides code control over individual actuating mechanisms or a group of them at the object through the controller of external memory units and the discrete signal shapers (ФДС).

This scheme works in two regimes: centralized and network data processing. In the former case the signals of the analog and digital sensors pass through the amplifier-shaper, data transmission multiplexor, analog-digital convertor, and controller of external memory units, are switched in digital code by means of the eight-bit microprocessor, go through

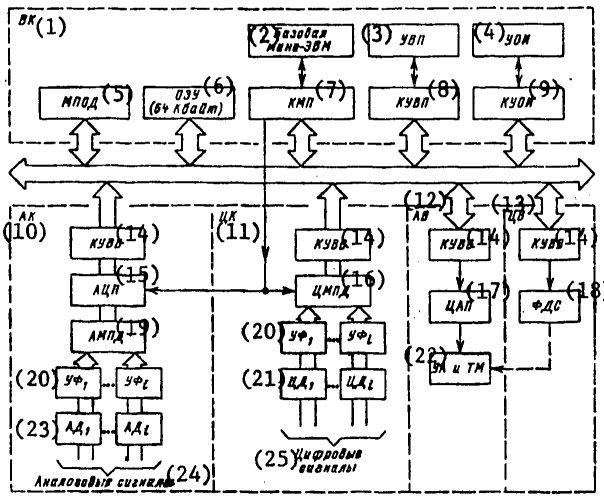


Figure 5.6.

Key:
(1) Computing Complex (BK);
(2) Base Minicomputer;
(3) External Memory Devices on Magnetic Disks and Tapes (УВП);
(4) Data Display Devices (УОИ);
(5) Microprocessor To Process Digital Information (МПОД);
(6) Main Memory (64 Kilobytes);
(7) Microprocessor To Control Input Channels and Links with Base Minicomputer (КМП);
(8) Controller for External Memory Units (КУВП);
(9) Controller for Data Display Units (КУОИ);
(10) Analog Input Channel (АК);
(11) Digital Input Channel (ЦК);
(12) Analog Output Channel (АВ);
(13) Digital Output Channel (ЦВ);
(14) Controller of Input-Output Units (КУВВ);
(15) Analog-Digital Convertor (АЦП);
(16) Digital Data Transmission Multiplexor (ЦМПД);
(17) Digital-Analog Convertor (ЦАП);
(18) Discrete Signal Shaper (ФДС);
(19) Analog Data Transmission Multiplexor (АМПД);
(20) Amplifier-Shapers;
(21) Digital Sensors;
(22) Automation and Remote Control Devices;
(23) Analog Sensors;
(24) Analog Signals;
(25) Digital Signals

initial processing in the 16-bit microprocessor, and then are transmitted by the eight-bit microprocessor to the base minicomputer for final processing. This minicomputer exercises analog or digital control over the object through the eight-bit microprocessor and the chain, controller of external memory units — digital-analog convertor — automation and remote control units or the chain,controller of external memory units — discrete signal shaper — automation and remote control units. In the latter case the 16-bit microprocessor performs complete data processing, exchanging with the distributed bank of information in the memory of other microprocessors for processing digital information by means of the eight-bit microprocessor.

Sensors are the most widely used measuring instruments. They provide information and most frequently are an essential part of industrial equipment.

Devices to filter the signals of the sensors are built into the individual channels and located right at the point of performance of the industrial operation. The amplifier-shapers are usually connected to the analog data transmission multiplexor by a short linkage, and the multiplexor may combine the functions of the amplifier-shaper and multiplexor of signals. The analog data transmission multiplexor employs analog keys which have a controlling digital input. These keys (IS series 590) pass an analog signal with an amplitude of ±10 volts through the resistances of the open channel (100 ohms) with a certain code in the controlling input. The controlling inputs for the IS series 590 are compatible by levels of logical signals with most domestic microprocessors used as input channel control units and also with the TTL [expansion unknown] elements from which most minicomputers are made. The amplifier-shapers may be connected to the digital data transmission multiplexor by longer linkages (coaxial cables up to 100 meters long) through mainline amplifiers based on series 559 integrated circuits (K559IP1-K559IP5). These integrated circuits provide transmission of digital signals with an amplitude of 0.2-2.4 volts from TTL elements on a matched cable with suspended load resistances (50 or 75 ohms) and contain several mainline transmitters and receivers. The eight-bit microprocessor controls the analog data transmission multiplexor through digital inputs and the digital data transmission multiplexor by special command (the eight-bit code makes it possible to connect 2^8 amplifier-shaper lines).

The digital signals go through the controller of external memory units, which are external microelectronic blocks with their own controlling memory (interface cards) that perform a number of control functions related to message transmission, and the microelectronic analog-digital convertor (for the analog input channel), to the "common line" interface devices and then to the 16-bit multiprocessor, the eight-bit multiprocessor, or the minicomputer. The microelectronic analog-digital convertor can be built with series 590 integrated circuits with suspended resistors that provide a threshold voltage to unlock the logical circuits.

The 16-bit microprocessor for processing digital control information is sometimes built into the industrial equipment to convert certain parameters (for example pressure) into others (humidity). In this case the multiprocessors are digital sensors that transmit the results of computations

through the digital input channel and 16-bit multiprocessor of the next level of control of the object. This is possible because SBIS's [very large scale integrated circuits] have been developed recently (the 8741 CB; East Germany) which contain on one n-channel metallic oxide semiconductor crystal plate the analog data transmission multiplexor, analog-digital converter, controller of external memory units, and an eight-bit microprocessor (a complexity of 280,000 elements). Such systems of very large scale integrated circuits may be mounted in a sealed case right on the object and can transmit signals to the microprocessor of the next level by the analog or digital input channel.

[111-11,176]

11,176
CSO: 1863

UDC 681.3

ASVT-M AND SM MINICOMPUTER SPECIFICATIONS

Moscow OSNOVY AVTOMATIZATSII UPRAVLENIYA PROIZVODSTVENNYMI PROTSESSAMI in Russian 1980 (signed to press 15 Oct 80) pp 230-231

[Tables 6.1 and 6.2 from chapter 6 of book "Fundamentals of Automating Control of Production Processes", by Vinfrid Kal'fa, Valeriy Valentinovich Ovchinnikov, Oleg Mikhaylovich Ryakin, Gans-Yurgen Sebastian and Vladimir Vasil'yevich Smirnov, Izdatel'stvo "Sovetskoye radio", 8,000 copies, 360 pages]

[Text] Table 6.1.

Specifications	Type of ASVT-M Minicomputer M-6000	M-7000	M-40	M-6010	M-400
Processor					
Length of Processed Words, bits	16	16	8; 16	16	8; 16
Permanent Memory	None	None	Commands	Micro-commands	None
Capacity, words	-	-	4,000	4,000	-
Cycle Time, microseconds	-	-	0.6	0.35	-
Word Format, bit	-	-	18	36	-
Performance Time for Basic Operations (with Fixed Decimal Point of Register-Memory Type), microseconds					
Addition	5	2.5	-	2.0	2.5
Multiplication	45*	5*	-	-	-
Division	60*	10*	-	-	-
Main Memory					
Capacity, kilobytes	8-64	8-256	1	8-64	8-128
Cycle Time, microseconds	2.5	1.2	2	2.5	1.2
Word Format, bits	18	18	18	18	18
Type of Control	Parity	Parity	Parity	Parity	Parity
Type of Linkage (Interface)	2K	2K	2K	2K	"Common Line"

* Through arithmetic expander.

[Table continued next page]

[Table 6.1 continued]

	Type of ASVT-M Minicomputer				
	M-6000	M-7000	M-40	M-6010	M-400
Number of Computing Installations Connected with One-Stage Addressing (Using Input-Output Expanders)	54	54	11	32	8*
KTDP [Channel with Direct Access to Memory]					
Number	2	2	0	0	0**
Speed of Data Transmission, kilobytes/sec	800	1,600	-	740 (by Microprogram)	1,600
Maximum Number of Computing Installations Connected through One KPDP	4	4	-	32	-

* Up to 4,095 with the use of wiring blocks

** The computer and memory modules are connected to the common line without a KPDP; its functions are given to the controller of the appropriate unit.

Table 6.2.

Specifications	Type of SM Computer SM-1	SM-2	SM-3	SM-4
Processor				
Length of Words Processed, bits	16	16	8; 16	8; 16
Storage of Microcommands				
Capacity, words	8,192	4,096	256	256-512
Cycle Time, microseconds	0.16	0.25	0.3	0.16
Word Format, bits	18	36	40	56(88)
Time of Performance of Operation with "Register-Memory" Type Fixed Decimal Point, microseconds				
Addition	2.5	2.2	5.0	1.4
Multiplication	36.6	10	-	10
Division	-	17	-	13
Main Memory				
Capacity, words	32,000	128,000	28,000	124,000
Cycle Time, microseconds	1.2	1.2	1.2	1.2
Word Format, bits	18	18	18	18
Input-Output Interface	2K	2K	Common line	Common line
Number of Connected Computing Installations				
Without Expanders	10	-	8	8
With Input-Output Expanders (or Wiring Blocks for the "Common Line")	55	55	Up to 4,095	Up to 4,095
Through RIM A714-5 Interface Expanders	1,728	1,764	-	-
Number of KPDP's	2	2	-	-
Maximum Input-Output Speed, words/second				
Through KPDP	250,000	700,000	800,000*	800,000*
Without KPDP	30,000	30,000	40,000	50,000

* The computer and memory modules are connected to the common line without a KPDP; its functions are given to the controller of the appropriate unit.

[111-11,176]

11,176
CSO: 1863

- END -