188026

JPRS-UCC-87-007

11 MAY 1987

USSR Report

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

19980610 071

SPECIAL NOTICE INSIDE

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

FBIS FOREIGN BROADCAST INFORMATION SERVICE

REPRODUCED BY
U.S. DEPARTMENT OF COMMERCE
NATIONAL TECHNICAL
INFORMATION SERVICE
SPRINGFIELD, VA 22161

DTIC QUALITY INSPECTED 6

26
A04

SPECIAL NOTICE

Effective 1 June 1987 JPRS reports will have a new cover design and color, and some reports will have a different title and format. Some of the color changes may be implemented earlier if existing supplies of stock are depleted.

The new cover colors will be as follows:

CHINA.....aqua
EAST EUROPE.....gold
SOVIET UNION.....salmon
EAST ASIA.....yellow
NEAR EAST & SOUTH ASIA...blue
LATIN AMERICA.....pink
WEST EUROPE.....ivory
AFRICA (SUB-SAHARA).....tan
SCIENCE & TECHNOLOGY.....gray
WORLDWIDES.....pewter

The changes that are of interest to readers of this report are as follows:

USSR reports will become SOVIET UNION reports.

The USSR REPORT: NATIONAL ECONOMY will be titled SOVIET UNION/ECONOMIC AFFAIRS (UEA).

The USSR REPORT: POLITICAL AND SOCIOLOGICAL AFFAIRS will be titled SOVIET UNION/POLITICAL AFFAIRS (UPA).

The following Soviet journals will be added to those which are already issued in separate series:

EKO: ECONOMICS & ORGANIZATION OF INDUSTRIAL PRODUCTION (UEO)
THE WORKING CLASS & THE CONTEMPORARY WORLD (UWC)
PEOPLES OF ASIA & AFRICA (UAA)
MILITARY HISTORY JOURNAL (UMJ)
FOREIGN MILITARY REVIEW (UFM)
AVIATION & COSMONAUTICS (UAC)
SOCIOLOGICAL STUDIES (USS)

If any subscription changes are desired, U.S. Government subscribers should notify their distribution contact point. Nongovernment subscribers should contact the National Technical Information Service, 5285 Port Royal Road, Springfield, Virginia 22161.

NOTE

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

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

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

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

PROCUREMENT OF PUBLICATIONS

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

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

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

Soviet books and journal articles displaying a copyright notice are reproduced and sold by NTIS with permission of the copyright agency of the Soviet Union. Permission for further reproduction must be obtained from copyright owner.

JPRS-UCC-87-007

11 MAY 1987

USSR REPORT

CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

CONTENTS

GENERAL

Problems of Improving Economic Mechanism of Computer Centers
(M. Kuter; VESTNIK STATISTIKI, No 10, Oct 86)..... 1

HARDWARE

Semitabular Calculation of Functions in Microprocessors
(G. V. Kantariya; SOOBSHCHENIYA AKADEMII NAUK GRUZINSKOY SSR, No 2, 1986)..... 8

Organization of Interprocessor Communications Channels in a Multiprocessor Computer System
(A. V. Kalyayev, S. Yu. Fomin; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 11

Methods and Means of Analog-Digital Modeling of Integrated Circuits
(V. V. Denisenka, N. I. Merezhin, et al.; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 11

Language for Description and Modeling of Parallel Network Control Algorithms
(V. V. Kuzmuk; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 12

SOFTWARE

Programming of Microprocessors in LYAPAS-M Language
(D. I. Cheremisinov; IZVESTIYA AKADEMII BELORUSSKOY SSR: SERIYA FIZIKO-TEKHNICHESKIKH NAUK, No 3, 1986)..... 13

Computer-Based Terminal Inquiry System for an Industrial Enterprise
(Yu. A. Shafrin, M. V. Rashkovskiy; MEKHANIZATSIYA I AVTOMATIZATSIYA PROIZVODSTVA, No 11, Nov 86)..... 19

APPLICATIONS

Automated Data Bank for Managing Financial and Credit Information
(Ye. R. Kochanova, A. V. Krasavin; MEKHANIZATSIYA I AVTOMATIZATSIYA PROIZVODSTVA, No 4, Apr 86)..... 27

Automation of Materials and Equipment Supply Accounting and Analysis Data
(T. N. Malinovskaya; MEKHANIZATSIYA I AVTOMATIZATSIYA PROIZVODSTVA, No 4, Apr 86)..... 33

Introduction of a Unified Generic System of Designation of Products and Design Documents
(N. S. Mishin; STANDARTY I KACHESTVO, No 11, 1986)..... 37

On Procedure and Rules for Presentation and Handling of Design Data on Magnetic Media Within YeSKD Framework
(Yu. B. Fratkin, A. S. Tretyakov, et al.; STANDARTY I KACHESTVO, No 9, Sep 86)..... 41

Examination of New Procedure for Development and Coordination of Technical Documentation at Regional Meetings
(STANDARTY I KACHESTVO, No 9, Sep 86)..... 48

Mixed Logic and Hardware Modeling of Digital-Analog Apparatus
(A. I. Petrenko, G. D. Kiselev, et al.; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 55

Book on Evolution of Auto-Wave Vortices Published
(Valentin Izrailevich Krinskiy, Aleksandr Berelyevich Medvinskiy, et al.; NOVOYE V ZHIZNI, NAUKE, TEKHNIKE: SERIYA MATEMATIKA KIBERNETIKA (EVOLYUTSIYA AVTOVOLNOVYKH VIKHREY) No 8, 1986)..... 56

Simulation Models and Methods of Making Decisions in Programming Harvests
(N. F. Bondarenko, R. A. Poleuktov, et al.; DOKLADY VSESOYUZNOGO ORDENA LENINA I ORDENA TRUDOVOGO KRASNOGO ZNAMENI AKADEMII SELSKOKHOZYAYSTVENNYKH NAUK IMENI V. I. LENINA, No 2, Feb 86)..... 58

THEORY OF COMPUTATIONS

Noiseless Coding of Combinatorial Sources, Hausdorf Dimensionality and Kolmogorov Complexity
(B. Ya. Ryabko; PROBLEMY PEREDACHI INFORMATSII, No 3, 1986)..... 63

Two classes of Algorithms for Decoding Based on Minimum Generalized Distance
(S. I. Kovalev; PROBLEMY PEREDACHI INFORMATSII, No 3, 1986)..... 63

Self-Adapting Algorithm for Minimax Nonparametric Estimation of Spectral Density
(S. Yu. Yefroimovich, M. S. Pinsker; PROBLEMY PEREDACHI INFORMATSII, No 3, 1986)..... 64

Quasianalogy and the Problem of Heat Conductivity
(Yu. M. Matsevityy; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 64

NETWORKS

Transmission of Packets by Blocked Nonmodified Random Multiple Access Stack Algorithm
(B. S. Tsyvakov, S. P. Fedortsov; PROBLEMY PEREDACHI INFORMATSII, No 3, 1986)..... 65

Deadlock Avoidance in Transmission of Information in Computer Systems With Programmable Structure
(E. A. Manakhova; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 65

Median Filtration of Deterministic and Steady Random Signals
(L. I. Piterbarg; PROBLEMY PEREDACHI INFORMATSII, No 3, 1986)..... 66

Structure and Interaction of Processes in Distributed Automated Data Bases
(I. N. Bacherikov; ELEKTRONNOYE MODELIROVANIYE, No 5, Sep-Oct 86)..... 66

PROBLEMS OF IMPROVING ECONOMIC MECHANISM OF COMPUTER CENTERS

Moscow VESTNIK STATISTIKI in Russian No 10, Oct 86 pp 35-40

[Article by M. Kuter, deputy chief of the Krasnodar Kray Machine Forms, Statistical Administration, Krasnodar Kray; candidate of economic sciences; in the section "Mechanization of Statistical Accounting Work" (title footnote *) (as a statement)]

[Text] The functioning of computer enterprises, specializing in computing and information and design work, must be based on stable organizational economic provisions regulating relations between the VTs [computer center] and the users.

Among the mandatory standards documents for use in computer centers are the "Unified Time and Performance Standards." These include the standards for work performed on both computer keyboard and perforating machines, and on new types of computer hardware (electronic accounting machines, electronic devices for recording information on perforated tape, magnetic tape and magnetic disks). Services rendered by using computers are paid for in accordance with price list No. U-01 "Rates for Computer Center Services" (1981 edition) under the single program mode of operation and the supplementary price list No U-01-1984/I which applies to multiprogramming. The amounts and cost of design work on developing software for user problems are established according to the "Type Time Standards for Programming of Problems for Computers." (footnote 1) ("Tipovyye normy vremeni na programmirovaniye zadach dlya EVM" [Type Time Standards for Programming of Problems for Computers], Moscow, GK SSSR po trudu i sotsialnym voprosam [USSR State Committee on Labor and Social Problems], 1980) In addition, internal departmental procedures and recommendations for determining the machine time of computers for software debugging and introduction, computing requirements for magnetic tapes and disks, and setting standards for consumption of paper media, etc. are also used.

In this article, we shall discuss the questions associated with organizing computer work accounting, and with determining the amounts, cost and prices of information and computer services rendered to the users.

The shortcomings of computer accounting by the price lists in effect have been covered in sufficient detail in the specialized literature. However,

some points have not yet been resolved. First of all, it should be noted that the price lists set identical rates by computer brands without taking the economic and technical characteristics of them into account, or specify the cost of a machine-hour of operation based on the balance cost, the increase of which by expanding the configuration does indeed affect work indicators, but does not always affect system throughput.

The YeS-1022, YeS-1030 and YeS-1033 computers initially came with 256K bytes of main memory, then 512K bytes, and the latter models of the YeS-1033 had 1M bytes with no change in the rate for services (75 rubles per machine-hour for the YeS-1022 or YeS-1030, and 80 rubles, for the YeS-1033), and with that the operating capabilities of the computers with various configurations differed substantially.

Rates per machine-hour of operation for the SM EVM [Small System computers] and the M 5000 VK [computer complexes] are set by a price list based on the balance cost of the computer. This leads to problems, for the solution of which an expanded set of peripherals is not required, being billed in different ways in different computers centers.

Developing price lists for VTs [computer center] services (for an hour of computer operation), apparently, should be based not on the computer balance cost (the cost of the units and devices which have replaced or which expand the set of computer equipment), but on the cost which takes the operating parameters of the complete set into account. Rates differentiated according to, say, the most critical resources (main memory, disk storage, printers) have to be set for identical types of computers (i.e. generalized by unification of the processor). Then the rate for operation of a computer, for example, the YeS-1033 with 512K bytes of main memory, three magnetic disk drives and two printers, will be somewhat lower than that for the same brand of computer with 1M bytes of main memory. The remaining set of peripherals is assumed constant for all sets of computer equipment. For the user, the increased rate will be compensated for by the reduction in the amount of computer machine time realized owing to the use of multiprogramming.

Additional devices, as a rule, are acquired by computer centers to ensure stable operation of computers or to solve individual (specific) problems, i.e. they are not required to realize the majority of jobs. The cost is included in the computation of the full rate for a machine-hour of computer operation (according to item 7 of the general provisions for a price list) only when the use of them is dictated by the conditions for solving a specific problem or allows a substantial reduction in machine time costs (processing cost).

The described principle for setting rates should also be spread to the Small System of Computers. The base configurations (models of machines) should be built based on the operating capabilities offered to the users of the computer resources. The existing practice, oriented to the computer balance cost, does not meet this condition and contradicts the requirement: equal prices for equal services.

For clarity, let us analyze the rate system for an SM 1600 equipped with the SM 1600.2620 and SM 2104.0506 single-type processors, the SM 1600.3510 main memory and the required minimum of mandatory peripherals. The rate per machine hour for the first model of this computer (balance cost of 136,300 rubles) is 20 rubles. The third model differs from the first by the expanded number of disk storage units by three units (total capacity varies from 42 to 84 megabytes) which has a positive effect on computer operation. However, the presence of 256 kilobytes of main memory and one printer does not allow a proper increase in throughput when processing data and the user is not compensated for the increase in the price for services because the rate is 50 percent higher (the balance cost of 170,900 rubles corresponds to a rate of 30 rubles).

Apparently, rates have to be set for first and fourth models (besides the additional disk units, it has an R-610 card reader and a printer), and in the other cases, operation of supplementary devices should be taken into account only when the user tasks require them.

Further development of organizational economic provisions for electronic data processing should be oriented to norms and standards for defining amounts of work, and the cost and price of services.

Such an approach allows ensuring methodological unity of computations for work performed on all types of computer hardware, setting identical prices for services equal in scope and use of resources, raising the scientific nature and objectivity of planning of amounts of computing and information work, expanding the functions of accounting and control, and ensuring the stability of relations between the computer centers, which perform the work, and the users.

For computer centers in the USSR Central Statistical Agency system, which specialize mainly in solving statistical accounting problems, it is not too difficult to determine standards for consumption of computer resources for most, if not all, problems.

In the organizational aspect of the effort on establishing standards for use of resources and determining the production cost and prices for computing and information services, there are three directions.

The first, the most widespread, consists in identifying the individual costs of the resources satisfying the conditions for organizing the production and technological processes of data processing in a specific computer center. Taken into account in the process are the composition and configuration of the computer equipment, the availability of the means and the prevailing method of data preparation, and the number and skill of the manpower. In some cases, organization of the technological processes and accordingly setting cost standards are predetermined by the quality of the source material submitted by the computer center users and the requirements imposed on the processing results.

Setting standards according to the use of individual resources, aimed, as a rule, at compensation for computer center costs, on the one hand, serves the stabilization of accounts with customers, which are maintained according to standard calculations, but on the other, does not promote development of computer work and often conceals imperfections in the technology and software.

In contrast to the first approach, the second presumes the use of common standards and rates, computed on the basis of them, for defined groups of problems for all categories of users (irrespective of their sector and departmental affiliation). The collection, "Rates for Computing and Information Services Performed by Computer Centers in the USSR Central Statistical Agency System," can be cited as an example. The collection was developed by the VNIPIuchet [All-Union Scientific Research and Design Institute for Accounting], USSR Central Statistical Agency, and is in the stage of an experimental check. The collection contains rates and fees for work performed by using various models of Unified System computers, the M 5000 VK [computer complex], and the Iskra microcomputer for automation of the individual accounting sections.

The amount of input and output information taken as a unit of computing and information services is 1,000 characters. Prices take into account the average cost of data preparation, the use of files and directories in the technological process, and the cost of machine media under the conditions of use of a specific set of SVT [computer hardware].

With this approach, computations for work performed are standardized, calculation of prices is limited to a common document, and the requirements of equal prices for equal services are met to the greatest extent. With that, the rate is independent of the individual organization of the process of data processing, and the configuration of a particular machine and the cost of it. Economic indicators of computer center functioning are directly related to the level of rationalization and optimization of technology which enables minimizing the expense for resources.

However, the common standards and rates do not take into account the specific features of computation of the same indicators (say, of accounting data) in the individual sectors of the national economy and, consequently, the different consumptions of data processing resources. Also not considered are the method of data acquisition and preparation of machine media, the number of copies of applications, level of quality the material submitted and a number of other factors which do not depend on the computer center. Of the number of sections included in the collection, the common standards and rates for processing of pension and insurance information are the most practical.

It should be noted that setting standards for the work for general customers in the USSR Central Statistical Agency system also requires improvement. Thus, the rate brackets for services by computer organizations to mechanize and automate accounting in agricultural enterprises, computed per employee with regard to average staff size, do not consider the number of document lines (indicators), falling within a calculation unit, which differ markedly by individual farms. The following table lists the data for processing

documentation on an M 5100 VK [computer complex] for the first section of a type design (labor accounting and payroll) by the individual farms in the Krasnodar Kray.

Table

Analysis of Monthly Volume of Information Processed as a Function of Number of Employees

Farm	Employees	Document Lines	
		Total	Per Employee
Novorossiysk Poultry Plant	330	12,280	37
Abrau-Dyurso Sovkhoz	714	10,700	15
Maykopskiy Poultry Sovkhoz	710	8,353	12
Krasnoye Znamya Kolkhoz	1,572	19,404	12
Sopka Geroyev Kolkhoz	1,968	19,945	10

In our view, the effect of two factors, the number of employees and the amount of information per employee, has to be assessed in setting rate brackets. Also, standards and prices should not exclude the factor of quality. Otherwise, all cost associated with poor organization and a low level of accounting work in user enterprises falls on the computer centers. A computer center (machine computation station) should have the right to increase the processing cost imposed on a customer as a function of the number of mistakes made by the user which are detected and corrected in the computer center.

The third direction in the use of norms and standards is oriented to a specific group of computer centers, most often in the same department, which use type software and technology, and standard forms for submission of source information for computations with identical quality of it. Examples are type design solutions, systems implementation of problems in sector ASU [automated control systems], and the systems for electronic processing of statistical information in USSR Central Statistical Agency computer centers, etc.

With this version, using technological standards per operation is more effective. The availability of standards for operation of user problems will allow determining the amount of work, organizing the computing process, and identifying the operations in which a computer center cannot use the standard specifications on consumption of computer resources. Setting standards by operation facilitates setting real prices for computer center services through specific standards and the commensurability of them with actual data on consumption of feedback information resources. For technological stages with a lengthy period of implementation, computations can be performed not only according to release rates for fully completed work, but also according to the standard resource consumption of individual operations, which as a result facilitates ensuring services actually performed match those included in the amount.

Setting standards by operation allows identifying, in the technological process of processing, the operations (input, checking, correcting), the execution time of which depends to a significant extent on the quality of the material submitted. And this, in turn, makes it possible to recompute standards for these operations with regard to the factor of quality. In processing KEOI [complexes for electronic processing of information] data on a computer, an average level of quality (and no worse) is considered, since the responsibility for the reports meeting the established requirements is borne by the personnel preparing and issuing the computer center statistical materials, i.e. the computer center in this case is both the user and the producer.

The experience of the computer system for the RSFSR Central Statistical Agency (footnote 2) (M. I. Kuter and V. P. Semenov, "Automated Maintenance of Standards for Consumption of Machine Time," VESTNIK STATISTIKI, No 3, 1984, pp 53-59) confirms the practicality and promise of organizing the setting of standards of computer resources and computation of prices for computing and information services by standard resource consumption on a wide scale. However, until recently, the legal question has remained open: Can norms and standards serve as a rate, or are just check functions inherent to them (i.e. services should be paid for according to actual expenses, but within the limits of the standards, or according to the standards without regard to actual consumption). In a number of works, in particular by O. Golosov (footnote 3) (O. V. Golosov, "Ekonomicheskoye stimulirovaniye sistemnoy obrabotki informatsii" [Economic Incentive for Systems Processing of Information], Moscow, Finansy i statistika, 1982), a no less important question has been raised: If the practice of computations according to the standard cost of processing is recognized as legitimate, then which agency should be the one which decides this action. The departmental approach is acceptable only for problems, in the solving of which a given department acts in the role of a user of the services of its own computer centers. The suggestion of conferring the role of experts in evaluating the scientific and technical level of programs, on the basis of which the account calculations are compiled, and also the quality of these calculations, on the territorial VTsKP [multiuser computer centers], which have the required skill potential and are free of narrow bureaucratic interests, is unconvincing.

For computing and information services rendered on the basis of interdepartmental agreements (for general customers of computer systems), common standards, and standards and rates per operation, approved by the directors of both departments, are quite acceptable. In developing norms and standards oriented to individual projects, the developer - customer - performer method is warranted. In this version, the operating and technical parameters of the software to be developed which meet the customer's requirements are established in a request for proposal. The standards for operation of the problems which are tested by the customer at the time of acceptance of the design and by the computer center (as the performer of the work) during trial operation are included in the engineering design. These or the revised standards are subsequently included in an agreement for industrial operation of the user's problems.

In setting rates for resources, no less an important role is allocated to the consumption of materials and the determination of the cost of them.

The shift to an industry of information processing without paper has predetermined the ways of reducing consumption of paper media for data entry and printing of the result. The different types of work being performed can be identified as a function of the computer center hardware configuration and classes of problems being solved on the computer: without use of paper media, with limited use of it for printing the results, and with the solution of statistical accounting problems in which the information printed is a legal document to be kept for a long time.

In connection with this, it seems legitimate to already now exclude the cost of paper for ATsPU [alphanumeric printers] and the inked ribbon for printers from rate per machine (standard) hour of computer operation. This will allow organizing payment for services for printing according to a special supplementary rate proportional to the resources consumed. Foreign experience indicates (footnote 4) (P. Keylingert, "Elements of Operating Systems," Moscow, Mir, 1985) that the existence of separate prices (rates) for printing forces the customer to more efficiently make use of resources in short supply or consume those not in short supply in place of them. Increased prices for each line printed allow achieving a substantial savings in paper.

When rates are set by the operation for consumption of paper for printers, a standard for technical printing (operating system messages, and solutions provided for by algorithm, if they are output to a printer) is established. Apparently, it is equitable to introduce such standards per job (module) which corresponds to the actual consumption of this resource.

In identifying paper consumption relations, a computation unit should be established (substantiated). That most acceptable is the amount of input information (in documents, indicators or other indexes) since this parameter is used to establish the norms and standards for consumption of other computer resources. The following alternatives are possible: paper consumption is unrelated to the amount of data being processed; resource consumption is linearly dependent on the number of units of input information; as the amount of data being processed increases and some value is reached, the number of result lines output to a printer per unit of source volume declines.

Placing magnetic media among the "inexpensive and non-durable items" identified in the separate group of "special tools and accessories" used for performing computing and information services allows, irrespective of the cost of objects in this group, using planned norms for depreciation of them which ensures uniform counting of the sums of depreciation of magnetic media (both disks and tape) by months throughout the entire period of use of them. The methodology of writing off the depreciation of PROM's for writing of programs for the Robotron 1720 EFBA should also be changed; this will promote ensuring objectivity in planning production costs.

These suggestions related to improving the economic mechanism of computer centers are aimed at raising the efficiency of the functioning of enterprises in the information processing industry.

COPYRIGHT: Izdatelstvo "Finansy i statistika", 1986

8545
CSO: 1863/73

UDC 681.32

SEMITABULAR CALCULATION OF FUNCTIONS IN MICROPROCESSORS

Tbilisi SOOBSHCHENIYA AKADEMII NAUK GRUZINSKOY SSR in Russian Vol 122 No 2, 1986 (manuscript received 25 Apr 1985) pp 277-279

[Article by G. V. Kantariya (presented by Academician V. K. Chichinadze, 24 Apr 85), Tbilisi State University]

[Text] This paper deals with a microprogram for semitabular calculation of a function and the efficiency of its use in microprocessors.

Algorithms for semitabular computation of elementary functions were presented in [1-3]. In this article, an example of microprogram implementation of this algorithm in microprocessors is shown. The function ln x was used as the example. It has maximal algorithm complexity because of the necessity of binary normalization of the operand. The microprogram corresponding to the algorithm, implemented as the instruction LOGN, is represented as a flowchart allowing use of LSI circuits with standard architecture. The algorithm for the operand x = P(1:7).X(0•2K) is represented by the relation

$$\ln x=\ln 16^{p} X(1:24)=(P-n)\,4\ln 2+\ln X1(0:k)+ \\ +2^{-k}\cdot X1(k+1:2k)/X1(0:k), \qquad (1)$$

where

$$k=12,\ X1=2^{n}X(1:2k),\ 1\leqslant X1<2,\ 1/4\leqslant n\leqslant 1.$$

To increase computation precision and reduce the ROM size for the logarithm tables, the operand mantissa is derived in the interval [1; 2), which produces the fraction n in the characteristic. The MP [microprogram] is written in the language described in [4].

LOGN Microprogram Flowchart

(Microoperations common to all operations and executed in parallel in the microprogram control unit are not indicated.)

```
L+01      R5(1:7),                          *characteristic and mantissa of
          R6(0:24):=X(1:7)³X(8:31);         operand are written to registers R5
                                            and R6:  specific type of addressing
                                            is a function of the program and
                                            computer architecture*.
02        JUMP IF ¬R6(0) TO L1;             *in status register when overflow
          CC [condition code]:=11;          condition code is set*.  END;
03  L1:   AMK:=SMK;                         *contents of microinstruction
                                            counter SMK is stored in return
                                            register--AMK [microinstruction
                                            address]*.
04        JUMP NOR;                         *go to the normalization MP
                                            [microprogram] which contains the
                                            return MK [microinstruction], just
                                            as other MP [microprograms]*.
05        R5:=R5-64;
06        R1(0:8):=LV1(R5);                 *binary shift left*.
07        R1(0:8):=LV1(R1);                 *multiply characteristic by 4*.
08        R6:=LV1(R6);
09        R1(0:8):=R1(0:8)-1,
          JUMP IF R6(0) TO L2;
0A        R6:=LV1(R6);
0B        R1:=R1-1, JUMP IF R6(0) TO L2;
0C        R6:=LV1(R6);
0D        R1:=R1-1, JUMP IF R6(0) TO L2;
0E        R6:=LV1(R6);
0F        R1:=R1-1;                         mantissa is derived in interval
                                            [1, 2) and characteristic is changed
                                            accordingly*.
10  L2:   APZU:=MK(25:28);                  *field in MK [microinstruction] as
                                            address of constant.*
11        R2:=RPZU;                         *number ln 2 is selected from PZU
                                            [ROM] of constants.*
12  UMN:  R3(0:31):=UMN[R2, R1];            microprogram for multiplication of
                                            ln 2 by 4 (P - n).*
13        JUMP IF ¬R1(0) TO L3;
14        R3:=R3-R2;                        *correction of product when
                                            multiplier in R1(0:8) is negative*.
15  L3:   APZU:=MK(25:28);
16        R5:=RPZU;                         *number 66 is selected as character-
                                            istic since integer of logarithm was
                                            shifted right by a byte.*
17  DEL:  R4(13:24):=DEL[R6 13:24),         *term X1(13:24)•2^-12/X1(0:12)
          R6(0:12)];                        is computed.*
```

```
18        APZU:=R6(0:12);                    *reference to table 1.  Since
                                             R6(0)=1 is mandatory, block 1 in ROM
                                             is allocated for storing
                                             ln[1+XI(1:12)].  Therefore number of
                                             entries into table 1 is 2^12*.
19        R4(1:24):=R4(1:24)+RPZU;           *function ln X1 is implemented*
1A        R3(0:31):=R3(0:31)+00*R4(1:24);    *number from R4 is transferred
                                             shifted right by a byte.*
1B        JUMP IF R3(0) TO L4;
1C        R3=⌐|R3+1;                         *translate into true code.*
1D   L4:  AMK:=SMK;
1E        JUMP NOR;
1F        R6(0:24):=R3(0:24);
20        CC(1:2):=R6(0).  R6(0)+10;         *set condition code to 01 or 10.*
21        KONETS [END];
```

The small NOR [normalization], UMN [multiplication] and DEL [division] microprograms can be written without looping (they contain about 10-15 microinstructions each). The LOGN microprogram requires a ROM with 12K bytes for table 1. The LOGN microprogram can be appreciably simplified when a specialized processor is used. Simple analysis of this fragment of the microprogram leads to estimating its complexity as 70 to 75 microinstructions. This number is slightly different from the number of microinstructions for the operation of multiplication with floating point. Also, the LOGN microprogram contains no calls to subroutines and its execution time is a function only of the normalization procedure. All this makes the efficiency of the method of semitabular computation of the function ln x evident. The matter is similar for other elementary functions.

BIBLIOGRAPHY

1. Kantariya, G. V., SOOBSHCHENIYA AN GSSR, 79, No 2, 1975.

2. Kantariya, G. V., SOOBSHCHENIYA AN GSSR, 80, No 1, 1975.

3. Kantariya, G. V., VOPROSY RADIOELEKTRONIKI, I, 1985.

4. Mayorov, S. A. and Novikov, G. I., "Printsipy organizatsii tsifrovykh mashin" [Principles of Organization of Digital Machines], Leningrad, 1974.

8545
CSO: 1863/15

UDC 681.323

ORGANIZATION OF INTERPROCESSOR COMMUNICATIONS CHANNELS IN A MULTIPROCESSOR COMPUTER SYSTEM

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86
(manuscript received 11 Jun 84; after revision 24 Oct 84) pp 14-17

[Article by A. V. Kalyayev and S. Yu. Fomin]

[Abstract] A study is made of an approach to organization of interprocessor communications in a multiprocessor system with structured implementation of computations, consisting of node processors with algorithms stored in firmware. The processors are designed to solve differential equations in partial derivatives. The approach described allows effective implementation of various interprocessor communications determined by the difference masks used in a multiprocessor system. Figures 5, references 4: Russian

6508/5915
CSO: 1863/139

UDC 681.333:38.-001.57

METHODS AND MEANS OF ANALOG-DIGITAL MODELING OF INTEGRATED CIRCUITS

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86
(manuscript received 20 Nov 84; after revision 18 Mar 85) pp 50-52

[Article by V. V. Denisenka, N. I. Merezhin and V. P. Popov]

[Abstract] Methods of analog-digital modeling of integrated circuits and examples of their implementation as specialized analog-digital computer systems are studied. Analog-oriented integrated circuit modeling methods have the advantage of greatly slowing transient processes for better analysis, but require large quantities of nonstandard equipment. Digital-oriented modeling systems, based on subdivision of the circuit being studied into fragments, significantly reduce integrated circuit analysis time by allowing parallel computation, increase modeling reliability, increase the maximum possible size of the integrated circuit which can be modeled and decrease computer memory requirements for modeling. The reliability of modeling with analog-digital computer systems is determined by the error of digital-analog and analog-digital conversion, while for numerical methods the error of mathematical

models is significantly greater. It is suggested that the analog portion of hybrid systems be constructed as an independent device to be used as a part of standard automated work stations. Figure 1, references 11: 9 Russian, 2 Western.

6508/5915
CSO: 1863/139

UDC 681.3.32:800.9.35

LANGUAGE FOR DESCRIPTION AND MODELING OF PARALLEL NETWORK CONTROL ALGORITHMS

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86
(manuscript received 29 Apr 84; after revision 2 Jul 84) pp 21-25

[Article by V. V. Kuzmuk]

[Abstract] A special language for parallel operation of multimicroprocessor control systems was constructed, meeting the following requirements: Simplicity of syntax, flexibility of design of parallel programs, capability of simultaneous processing of several program modules, adequacy of program modules to structural elements of the base algorithm, and simplicity of rules for parallel operation of programs. The language is called YAPAUSS, a Russian acronym for 'language for parallel algorithms using control networks.' It can be used for algorithmic description of parallel interacting processes, and is suitable for construction of a system of transitions or macrotransitions, as well as control of operating modes in selection and distribution of labels. Figures 2, references 15: 11 Russian, 4 Western.

6508/5915
CSO: 1863/139

UDC 681.3.068-181.4

PROGRAMMING OF MICROPROCESSORS IN LYAPAS-M LANGUAGE

Minsk IZVESTIYA AKADEMII BELORUSSKOY SSR: SERIYA FIZIKO-TEKHNICHESKIKH NAUK in Russian No 3, 1986 (manuscript received 10 Dec 85) pp 103-107

[Article by D.I. Cheremisinov, Institute of Technical Cybernetics, Belorussian SSR Academy of Sciences]

[Text] Discrete devices constructed on the basis of microprocessor equipment are customarily divided into two major classes as a function of their area of application: microprocessor systems designed to replace traditional computer equipment; and built-in devices designed for the purpose of implementing a specific algorithm and representing part of a system containing other components. The algorithm implemented by the device defines the dependence of its output signals on the signals supplied to its inputs. A widely used example of a built-in microprocessor system is devices for controlling an automatic production-process machine (an automatic machine tool, an assembly line and the like). Microprocessor systems of type a), for the sake of convenience, we will call microcomputers, and of type b), controllers. A distinctive difference between them is the fact that controllers are devices having a program written into the ROM and which cannot be changed during use.

Components such as registers, flip-flops, gates, multiplexers and the like have served as the traditional basis for designing built-in systems. With this, the algorithm implemented by the discrete device depended on the configuration of component connections. Since in a microprocessor system the functioning algorithm is determined by the program placed in the system's memory, then one and the same controller can be used, for example, in an assembly line and a home washing machine. Thus, the use of controllers makes it possible to unify the equipment of built-in systems. The designing of these systems boils down to programming the controller.

The situation existing at the present time in the data processing field is characterized by a tendency toward lowering the cost of computer hardware with a simultaneous increase in the cost of software. Computing (microcomputers) and control (controllers) equipment developed on the basis of microprocessors is the most popular and inexpensive class of computers. The problem of an efficient programming method is especially topical for them.

The specific nature of the programming of microprocessor systems, in particular of controllers, as compared with traditional computer equipment, involves changes both in the range of problems of the tasks confronting the programmer, and in the programming method. In the process of preparing programs for traditional computer equipment, it is important to use precisely the computer on which the programs will be run. This approach is impossible for controllers because of technical reasons (the absence of a RAM for programs). In this connection, engineering computers with the appropriate programs (programming cross systems) are widely used in the programming of microprocessors.

Another (more important) difference in the programming approach for microprocessors from the traditional involves a change in the role of the programmer in the development of programs. Within the framework of the traditional approach, a kind of division of labor was formed whereby most often the programmer's work consists in designing a program for a type of computer specified in advance, for a finished formulation of the problem. In the development of a built-in system, the process of designing and implementing the controller's programs must be closely tied to the designing of all the rest of the system. This in turn necessitates the participation of the programmer in development and formulation of the problem and of the architecture of the microsystem to be used (he must do the work of the analyst and systems programmer in the traditional approach). The traditional division of labor is usually not feasible for economic reasons: A great number of people are not required here, since the microprocessing unit as a rule constitutes a small part of the system.

The role of the programming language has changed because of differences in the program preparation processes. In the traditional approach the principal requirement for it is that it must provide an efficient interface with the computer. The programming language for microprocessors, because of the fact that the programmer himself must develop the formulation of the problem, must be a thinking tool to a much greater degree than before. Under conditions of incomplete information on the hardware configuration, the programmer often cannot count on the existence of a compiler and must himself take care of implementation of the language to be used. Thus, the programming language for microprocessors must on one hand be the language of the theory within whose framework the problem formulation is worded and, on the other, provide an efficient interface with the hardware and permit simple implementation.

In programming microprocessors in the LYAPAS-M language [1] it is possible to attain all the goals set, although they appear mutually exclusive. LYAPAS-M belongs to the same class of languages as BCPL [2] and C [3], which are usually called middle-level languages. Their principal feature is the high quality of object programs, which is comparable to that which can be achieved in programming in an assembly language.

LYAPAS-M is preferable to languages of the assembly language type for reasons which it is possible to call engineering, but they are of fundamental importance. First, the representation of a program in LYAPAS-M is much more compact than in an assembly language. A single instruction of an object program in

an assembly language is represented by a character string, whereas in LYAPAS-M it is represented on average by a single character. LYAPAS-M surpasses many high-level languages in terms of the compactness of programs. Secondly, the syntax of LYAPAS-M is simpler than the syntax of an assembly language.

The compactness of the representation of programs and the simplicity of the syntax are factors which considerably simplify implementation of the language. Besides, this makes it possible to organize the compiler so that it works without accessing an external storage. For these reasons LYAPAS-M compilers construct a program an order of magnitude faster than an assembler. Thus, it can be assumed that LYAPAS-M provides an efficient interface with the computer and permits simple implementation.

In order to use LYAPAS-M as a thinking tool, its expandability can be utilized. Although growth of the language is made possible by traditional means, in the expansion mechanism of LYAPAS-M there are important elements which contribute to greater streamlinability of this mechanism as compared with other languages. The introduction of a new operation into the set of operations of an idealized LYAPAS-M machine and the substitution of this machine by another can be employed for the purpose of expanding LYAPAS-M.

An idealized LYAPAS-M machine is a device whose behavior describes the sense of the language's operations. It has, as do real computers, a two-level memory (working and file storage), is controlled by a program in LYAPAS-M, and is so constructed that for the purpose of expanding its operation set it is sufficient that as of the moment of execution of a new (macrooperation) there be in the file a routine implementing it and represented in LYAPAS-M. Compilation of it is performed automatically in execution of the macrooperation.

In many programming languages a similar operation requires specifying the procedure implementing a non-elementary operation in the text of the program in which it is used, or obligatorily placing the procedure in the operating system library before running the program utilizing it. The difference of the method of expanding the operation set in LYAPAS-M is that, first, in the execution of macrooperations it is possible to use programs represented in the source language, and not in the internal, as usually is the case. Secondly, it is sufficient that the implementing program be available as of the moment of the execution of a macrooperation, and not as of the start of the execution of a program utilizing a non-elementary operation.

The substitution of an idealized machine makes it possible to implement expansions syntactically different from the base language. Usually in making the substitution a precompiler is used, whose purpose is to cause the LYAPAS-M machine to simulate the behavior of the new one. The implementation of LYAPAS-M expansions by means of preprocessor processing is highly efficient because of the good quality of the object programs of the basic compiler. Thus the principal obstacle, present in traditional languages, to employing this method is eliminated. It consists in the fact that the quality of an object program in two-stage compilation drops in proportion to the product of the quality factors characterizing the precompiler and basic compiler. A

LYAPAS-M compiler makes possible a quality of the program produced as the result of two-stage compilation close to the quality of the program generated by the precompiler.

It is relatively simple to develop a precompiler implementing the substitution of a LYAPAS-M machine, since the operation of substitution of the machine can be expressed in the language. The entire process of the processing of programs (precompilation, compilation and execution) is programmed in LYAPAS-M without enlisting the facilities of the operating system.

The employment of expansion facilties is an approach to the development of a LYAPAS-M programming system in the direction of an ideal one. In this programming system, only specification of the problem to be solved is required of the programmer in the development of a program. There is sufficient basis for assuming that this ideal is unattainable [4]. We approach it in a real system if for the purpose of formulating problems from a certain field a language convenient for this is used, and there are in the system facilities for transforming problem specifications into programs. Here by convenient is meant a language which efficiently performs the function of a thinking tool in a certain problem area [5]. The set of tools for the development of software for microprocessors [6] offers the programmer facilities for expanding LYAPAS-M by practically any set of symbols. The BESM-6 serves as the basic computer of the set of tools.

Let the programmer have a certain set of symbols convenient for use as a problem specification language. In order to use it as a LYAPAS-M expansion the programmer must design a machine whose behavior defines the sense of the symbols introduced (e.g., as suggested in [7]), and use it in place of the LYAPAS-M machine. The LYAPAS-M expansion called the Regular Expression Language (YARV) [8] can be used for specification of the precompiler design problem which thereby arises.

The program in YARV implements the operation of transforming character strings. It consists in analyzing the source string for the nesting of substrings with a specific structure and in constructing a new string by the substitution of substrings found in the source string by other ones. The procedure for analyzing the source string is the same as in the SNOBOL language, and is specified by indicating the model, of which a regular expression serves as the syntactic form. The experience of using YARV has demonstrated that the program representation space is reduced by a factor of three to five as compared with representation in the base language.

For the purpose of programming controllers it is convenient to utilize languages for describing discrete automata, such as sequential, algorithm-logic-circuit, regular-expression-system, parallel-control-algorithm, and the like. The machines which define the operating semantics of these languages are relatively uncomplicated [7]. In this connection, from the viewpoint of simplicity of implementation, for them it is feasible to develop direct compilers for the instruction codes of individual controllers. However, implementation in the form of a LYAPAS-M expansion is more convenient for the purpose of eliminating semantic errors in debugging.

For the purpose of automating the development of compilers, a system for designing them has been devised which includes YARV and the SIGULL [SEAGULL] instruction generator programming language [9]. The instruction generator, which constructs the object program in the compiler, is described in SIGULL in the form of a set of tables describing mapping of the operations of the source language machine into the instructions of the object microsystem. Another task of the system consists in implementation of LYAPAS-M compilers.

Microprocessor system debugging facilities can also be developed by expanding LYAPAS-M. Their hardware rarely provides good capabilities for debugging; therefore, it is a good idea to employ models of hardware for the purpose of searching for errors in programs. Although software models of hardware (emulators) are of low precision, their employment makes it possible to simplify debugging, since in this case it is easy to trace the behavior of "motored" programs. In programming microprocessors in LYAPAS-M, an expansion constructed on the basis of the CDL language [10], which is called the Register Copy Language (YARP), can be used for the purpose of specifying the emulator.

The typical conditions for the development of a discrete system for which a set of tools has been designed consist in the following. Let the controller hardware exist and let it be known that the algorithm for forming the implementation for which it is intended belongs to the class of those which can be represented in a certain language, L . The design problem consists in designing a program for the controller from a specific representation of the algorithm in language L . The following ways of solving it are possible:

a) If a compiler exists for language L for the specific controller, then the design problem is solved most simply and automatically--by using it.

b) If there is no compiler for language L , then it is possible to convert manually the description of the algorithm to be implemented from language L into one of the languages for which a compiler exists, and to design the required compiler.

If it is kept in mind that the development of a built-in system is part of the process of designing the enveloping unit, then the employment of the latter method makes it possible to distribute optimally the load on programmers over time, and in many cases to reduce the labor intensiveness of programming. The development of a programming automation system performed during the period from the start of designing of the enveloping unit to the development of the engineering assignment for the built-in system makes it possible to effectively make the work of programmers parallel to the work of other developers. Besides, with the state of the art of compiler design theory this work is in many cases less labor intensive than the programming of a long applications program. After drawing up the requirements for the built-in system to be designed, converting the formal specifications of the design assignment into a program does not require considerable input as a consequence of the effort spent on the automation of programming in the preceding step.

The effectiveness of the facilities, contained in the set of tools, for tailoring to the language and the microprocessor has been confirmed experimentally by the development of several compilers within short amounts of time.

BIBLIOGRAPHY

1. Zakrevskiy, A.D. and Toropov, N.R. "Sistema programmirovaniya LYAPAS-M" [LYAPAS-M Programming System], Minsk, 1978, 220 pages.

2. Richards, M. PROC. AFIPS SJCC, Vol 34, 1969, pp 557-566.

3. Ritchie, D.M. et al. BELL SYSTEM TECHNICAL J., Vol 57, No 6, 1978, Part 2, pp 1991-2019.

4. Lavrov, S.S. KIBERNETIKA, No 6, 1982, pp 11-16.

5. Iverson, K. COMM. ACM, Vol 23, No 8, 1980, pp 444-465.

6. Cheremisinov, D.I. USIM, No 5, 1981, pp 68-71.

7. Cheremisinov, D.I. "Avtomatizatsiya analiza i modelirovaniya logicheskikh setey" [Automation of Analysis and Modeling of Logic Networks], Minsk, 1981, pp 71-85.

8. Cheremisinov, D.I. PROGRAMMIROVANIYE, No 5, 1981, pp 30-37.

9. Cheremisinov, D.I. PROGRAMMIROVANIYE, No 3, 1982, pp 44-51.

10. Chu, Ya. "Organizatsiya EVM i mikroprogrammirovaniye" [Computer Organization and Microprogramming], Moscow, 1975, 592 pages.

COPYRIGHT: Vydavetstva "Navuka i tekhnika" Vestsi AN BSSR, seryya fizika-tekhnichnykh navuk, 1986

8831
CSO: 1863/33

UDC 681.3.06

COMPUTER-BASED TERMINAL INQUIRY SYSTEM FOR AN INDUSTRIAL ENTERPRISE

Moscow MEKHANIZATSIYA I AVTOMATIZATSIYA PROIZVODSTVA in Russian No 11, Nov 86, pp 35-38

[Article by Yu.A. Shafrin, candidate of physical-mathematical sciences, and M.V. Rashkovskiy, engineer: "Computer-Based Terminal Inquiry System for an Industrial Enterprise]

[Text] The introduction of remote processing facilities in ASU [management automation systems] of enterprises brings the systems of organization and economic management to a new qualitative level, where it is possible to switch over from batch information processing to immediate communication between the terminal users and the computer. This greatly improves the immediacy and reliability of administrative enterprise management, providing feedback for functional services with the mainframe in a conversational interaction mode and thereby improving the quality of managerial decision-making.

High-power universal data base management systems [DBMS] such as OKA, SETOR, BANK, INES and others have been introduced widely recently. Some of these have their own remote processing monitors; others operate with commercially available systems such as KAMA, KVANT and others.

On the other hand, it still remains difficult to choose an optimal base information system for a medium-scale enterprise with limited resources in terms of the hardware configuration and composition of the service personnel. The main difficulties include the following: the complexity of design and reorganization of hierarchical and network structures; the difficulties with the introduction and operation of a system involving the solution of a wide range of problems from effective software to simplicity of terminal user communication; and the need for the user to develop by its own forces on the basis of a remote communication montior a specific system of remote data processing or terminal inquiry at the level of application software, as well as other problems.

The general-purpose systems have shortcomings which manifest themselves as difficult servicing (OKA DBMS), restrictions on the types of data that can be retrieved unless connection are established between such data previously

at the stage of system design, stringent requirements for RAM capacity, terminal operation regimes, etc.

At the same time, the functional redundancy of general-purpose systems, which is usually described as a merit (such as the adaptability to various types of terminals, including remote units, the restart of uncompleted queries, the interactive language, which, however, is often abandoned by the user because of its complexity, etc.), are often not needed by the end user, who for a long time had access only to a limited set of hardware and operated with narrowly specialized queries.

It is beyond doubt that a large number of local information systems for organizational-economic management could be more efficiently constructed on the basis of specialized data processing systems [SOD], which strictly are DBMS (footnote 1) (Ye. P. Velikhov, "Data Banks: The Base for Computerization," EKO EKONOMIKA I ORGANIZATSIYA PROMYSHLENNOGO PROIZVODSTVA, No 2, pp. 162-166, 1985).

The terminal query system [STO] for an YeS computer described in this article has been developed on the basis of the simple relational data base ROIN [Relational Organization of Information]; in contrast to general-purpose complexes, it is ready for practical operation in a management AIS [automatic information system] with the use of YeS-7920 local display terminals.

A Brief Description of SOD ROIN

The relational data base ROIN is a set of arbitrary access data stored on one or several magnetic disk packs. The data set is a combination of independent logical files (relations) of a tabular structure linked by a common catalog included in the same data set.

Each logical file [LF] is a two-dimensional table consisting of a fixed number of formated records (tuples) of a similar structure; each record contains a fixed number of numeric or textual data units (data elements). Each LF has a unique name up to five characters long. The data base can contain up to 400 logical files.

Each data unit is defined by a stencil specifying the data name (up to five characters), its type and length. The set of data stencils (from 100 to 200) included in a record forms the logical file format and is created by the data base developer.

For example, when compiling a list of products of a nonferrous metal processing plant one can include the following data:

(1) the product code as per All-Union Classifier with the name ZYX, length 3 and the type "number";

(2) the name of product with the name of NMIZ [not further identified] with the length 20 and the type "text."

Each particular copy of a record (tuple) of such a file can contain, for example, the following information: 445 copper pipes.

A given ZYX of the same type and length can be included in several other LF. For example, the guide of technical-economic indicators, the portfolio of invoice-orders, etc.

The format indicates the main key and length of an LF in a record; the number of records may not exceed 1 million. When the format is included into the catalog, the logical file is automatically formated with "empty" records (containing no information). Processing of a file consists in reading and updating records; these processes substitute real records for "empty" ones and "empty" records for real ones ("removal" of a record). In addition, information inside real records can be modified; the total number of records in an LF remains constant.

All application programs are run in batch processing or conversational mode controlled by the main monitor of the system, which operates as an application program of OS within a single item of a job.

The status of application programs is assigned also to numerous system modules which handle all the necessary service functions: LF loading and modification, editing, LF presentation for a printout or terminal display, sorting, etc.

The relational approach, which does not establish any routing connections between data, provides an arbitrary access to information with the aid of both relational algebra (projection and linkage of relations) and relational calculus. A simple reorganization of DB is provided (modification of LF format with retention of the contents, addition and removal of LF).

One application program of the main monitor is the terminal inquiry monitor [MTO], which controls the application models of STO.

The Operation Principles of STO

The terminal inquiry monitor is a program providing the concurrent operation of up to 32 local alphanumeric YeS 7920 terminal displays assigned to the individual users (subscribers). Each terminal is identified by a serial number (1, 2, ..., 32) and can be installed, for example, in the VTs [computer center], in the factory manager's office, in a workshop, etc. (within up to 1200 m from VTs).

Any user query for extraction, updating, processing and representation of information is performed by a particular application module (built into the system or developed by the user); the module is loaded, initiated and removed from the memory by the MTO monitor.

STO ROIN has some important features by which it differs from general-purpose remote processing systems.

1. MTO operates as a single job of an assignment item initiated by the main system monitor. This means that the subscribers are served sequentially (one after another) as the system passes around all the terminals in a cycle.

Special methods of compulsory program quantization are provided which ensure the regular transfer of control from one terminal to another, simulating a time-sharing mode. This approach has the following advantages:

the total service time of the users is reduced considerably (there are no subtasks and the number of references to the supervisor is reduced);

a constant and effective tracking of the activities of application modules [PM] is provided;

the RAM capacity that is required is small; and

the system is compact in size.

STO operation experience indicates that, although the system has certain shortcomings (the possibility of uncontrolled delays and emergency stop of the entire system in case of error in any single PM), the operation is not impaired because of effective measures of protection built into MTO.

2. STO offers the user a well-defined procedure for subscriber-DB communication, allowing the construction of a broad class of question-answer systems and systems gathering organizational-economic data. This means that it suffices to devise a logical file and create a direct data set (it is not even necessary to load the files, except for the terminal inquiry file).

To initiate operation one can use system modules for loading, reading and editing the files and producing formal references.

All application modules are connected into the terminal query file [FTO], which is an ordinary LF of the same DB.

Each record in FTO describes a particular PM and contains the following data items: the name of the application module; the name of the attention key (if necessary); the admissible time of stay in the RAM; mnemonics, user scale, etc.

The user scale posts for each terminal the permission code, which serves to protect the application module from unauthorized access from the display of a particular terminal.

When the system is started specific tuning parameters can be indicated with control information.

After the terminal query monitor is started a title page of the system is displayed on the screens of all terminals which are switched on: title; field for information setting (the beginning of the field is marked by an X, underscored by the cursor); and a list of functions (application modules) which the particular user can request specifying the necessary service data.

Three methods of query by the user are provided: keying the module name (for example, FORS); keying the serial number of the module (for example, 12); and depressing their appropriate attention key (e.g., PF8) if one is specified in the list of functions.

After a module is summoned it is loaded into the memory. The results of execution displayed on the screen in some form are the answer to the query. After the operation is stopped (whether automatically or upon user's request), the title page is displayed again and the results of module execution are indicated in the header. The user can then enter the next query.

The header also contains short messages from other terminals, rejections with indication of reasons, etc.

When there is not enough space in the RAM for the application module, the latter is placed on a queue and executed after the memory is cleared. The user can cancel a delayed query.

The operation of STO ROIN showed that since the application modules are not large (from 10 to 10-12 kilobytes), while the total memory of the system is 200-240K, the delay mechanism is hardly ever triggered (a maximum of 8-10 terminals operate simultaneously at a medium-sized enterprise).

There are no restrictions on the particular functions of the application modules. Some of the application modules are of a general system purpose and can read and edit LF information, produce formal references, etc. There is also a FOTO module, which enables the support group to take a "snapshot" of the system state at any instant, specifying both the state of each of the terminals (on-off) and the actions of the subscribers.

The user in compiling an application module defines its functions and protocol and communication with it (in particular, the composition of the control information, the internal conversation scenario--"menu," "prompts," etc.). If necessary, the application module can provide a password protection of its functions from unauthorized access in addition to the system protection provided in the FTO file.

The speed of query answering depends on the particular function performed by the module and varies from a fraction of a second (with direct access to reference data) to minutes (in case of a high level of activity and the processing of long logical files).

There are two external scenarios of application module operation providing the choice of an optimal processing protocol:

the module is summoned, executes its functions and returns control to the remote monitor, which produces the title page for a new query regardless of the user's demands; and

the module can stay in the memory for an indefinite time, performing the user's query according to the internal scenario (e.g., the user can scan or edit a file).

In the latter case other queries may experience a shortage of memory capacity, resulting in delay of their execution. Depending on the specific operation conditions (available RAM capacity, module functions, module size, user priorities, etc.), the support group can limit the time of stay of a particular module in the memory, specifying in FTO, for example, 5 or 120 s. Upon expiration of this time the module is automatically excluded from the memory and the corresponding message is sent into the title page header. A PM, such as the system surveillance module FOTO (4K) can be kept in the memory permanently, and by pressing the enter key the user can receive a snapshot of the system.

All the diagnostic information concerning the STO functioning is sent into the operation protocol of the system, which registers all user queries and results of their execution, the state of hardware malfunctions, program interrupts, instances of memory shortage, messages to another terminal, etc., indicating the clock time of these events.

If the user turns on his terminal while the system is in operation, the title page will appear with a delay (from 1 to 30-40 s) depending on random causes and MTO tuning parameters. There is a possibility of excluding any terminal from the system during the course of operation and returning it into operation again (for example, after correcting a malfunction).

The RAM capacity is determined by the size of the MTO with the main monitor (some 54K) plus the sum of all PM kept in the memory simultaneously. The system remains serviceable if the memory is not smaller than 54K plus the size of the maximum module to be executed.

Programming of Application Modules

In contrast to general purpose systems, STO ROIN is characterized by a complete process of PM development, which is largely based on the flexibility of the relational method and the simplicity of projecting the relation links. The concept of external processing of logical files and internal processing of lists, each of which is a mapping of a relation in the working memory, is widely used.

Most standard procedures (reading and updating LF, internal sorting and searching, operating with the terminal screen, etc.) are carried out by internal system modules, some of which are built into the remote monitor, while others are connected to application modules during editing.

There are three programming levels: programmer level (Assembler language), user level (data manipulation language ROIN-2) and end-user level (formal query language based on elementary notions of relational calculus).

The data manipulation language ROIN-2 is a simplified Russian version of BASIC language, supplemented with simple operations of relational processing of LF and internal lists and also operators for interacting with the display screen. By way of convention, ROIN-2 operators can be divided into three groups:

commands of information processing and output, and control commands which have analogs in BASIC (arithmmetic operator, IF, GO, format print, cycle, go to subroutine, call module, etc.);

relational processing commands operating with LF and lists as a whole or with separate tuples (open LF or list, read LF or list, enter LF into list, include, update, find, etc.); and

commands for operationg with the screen (produce screen, read screen, print screen).

When the system operates with a terminal the screen is represented in the RAM by its "image" of 1920 characters (according to the number of positions on the screen). The programming is reduced to displaying the "image" onto the screen [VEKR], the mapping of the screen contents into the memory [ChEKR], the printing of screen (or, more correctly, the screen "image") and, if necessary, manipulation of the contents on the screen with simple transformation operators (number-text, text-text and text-number).

The programmer can use the addresses of screen positions and the cursor defined by a pair of numbers, "line number-position number," and the textual form of attention signal (1,2,...). With this concept the requirements for programmer skill are minimized, as is the likelihood of semantic errors in the module.

DB information is accessed by LF names and data item names, so that application programs are largely independent of changes in DB. More exactly, the independence is complete if the data item is not moved to a different logical file.

An important feature of ROIN-2 language is the complete system of PM adjustment, which, in addition to the oridnary syntactical control, makes it possible in case of a semantic programming error to produce a compact dump of the module state specifying the diagnostic data of all of its entities (variables, LF, lists, cycles, etc.).

The end user level is represented by the formalized query language ROIN-3, based on the elementary principles of relational calculus; it is serviced by the application system module FORS (formal reference).

After this module is called the subscriber sets on the screen the query for information using data names and symbols > (greater than), < (less than) and = (equals). The query is stated as follows: Select from the set of real records of a certain LF a subset in which the values of specific data satisfy (or do not satisfy) conditions specified in the query. In addition, the query can indicate which of the data from the selected subset must be added up. As many as 60 conditions can be specified.

For example, the data base can have a shipping file with the name OTGR [shipping]. In addition to other data items, this file contains the following units: workshop number TsEKh [workshop], product code ZYX and the amount of product shipped KOL [amount] (in kg). The subscriber wants to know the amount of copper pipes (ZYX = 445) shipped by workshop 12.

The query has the following format: OTGR, KOL: TsEKh = 12; ZYX = 445.

At the level of the end user the programming is reduced to compiling on the terminal screen the text of an arbitrary query in the language of relational calculus.

Despite the limitations of this method the language has been widely used by managerial staff of enterprises in developing estimates as part of organizing the flow of orders at the plant (the amounts and types of articles received), evaluating the course of plan fulfillment by the enterprise units from various points of view, etc.

On the other hand, experience suggests that users (especially at higher levels) usually avoid stating even elementary queries in terms of relational calculus. In view of this, STO ROIN provides a capability of placing the frequent queries in a special DB file named BAZAP [query bank]; after this the query can be called by keying in a three-character identifier with a service feature; e.g., *AOI.

These queries function as application microprograms; the arbitrary identifier is the name of a microprogram.

At a nonferrous metal processing plant where STO ROIN has been put into operation a different form of query has been practiced where the user fills in the data value into a reference stencil flashed on the screen.

Generally, the operational convenience of STO ROIN is based on reliability and high speed of arbitrary access to information and simplicity of system installation and operation. Practical estimates show that while it takes 1-2 years for a medium-skilled user to learn how to operate a general purpose system, he can master STO with SOD ROIN in 1-2 months.

COPYRIGHT: Izdatelstvo "Mashinostroyeniye," "Mekhanizatsiya i avtomatizatsiya proizvodstva," 1986

9922
CSO: 1863/137

AUTOMATED DATA BANK FOR MANAGING FINANCIAL AND CREDIT INFORMATION

Moscow MEKHANIZATSIYA I AVTOMATIZATSIYA PROIZVODSTVA in Russian No 4, Apr 86 pp 41-43

[Article by candidates of technical science Ye. R. Kochanova and A. V. Krasavin]

[Text] Banking automation is now undergoing rapid development, and in association with this the problem of increasing the efficiency with which existing technology is used and creating new methods for managing and computing banking data becomes especially acute.

Increasing the efficiency of client monetary and credit service is inconceivable without introducing data storage and transmission systems and creating automated banking data management methods by studying leading foreign experience.

Work is being conducted within the member nations of the Council for Mutual Economic Assistance to organize data banks in the financial and credit system. The Hungarian National Bank (VNB) has specialized in the creation of an automated data bank (ABD) within the framework of an international conference on the application of computer science in socialist nation banks.

In connection with this, the experience gained by the VNB in creating a currency exchange data bank that occupies a central place in the data processing system of the VNB is of interest. It operates in accordance with data base management system design principles. The following management levels should be distinguished: Presentation of data for making decisions in an interactive mode (highest management); presentation of data for solving immediate tasks (middle management); data control.

We shall examine ABD operation at all levels. The "Honeywell" 66/20 system, which includes a computer having four 100 Mbyte disks and magnetic tape storage devices (7 and 9 tracks), comprises the hardware. Six terminals can be connected up to the computer.

Data are entered into the computer in two steps: At the first step base data (which are fundamental and stored for long durations) are generated and

entered; at the second step the immediate or variable data that define the state of an entity at any given moment are entered.

Thus, the data are divided into fundamental (base) data that are stored in the computer and comprise the data base, and non-fundamental (variable) data which enter the bank on a daily basis. The following information is entered into the data base in order to solve credit tasks: The credit balance indicators of a nation; VNB credit balance indicators; credit data; requirements for the statement of and data pertaining to the investments of economic organizations; data regarding funds allocated for savings, the financial and technical details of investments, assets, and money circulation.

The data base is updated every year; data pertaining both to the current year and to the preceding four to five years are stored.

The creation of a special identification system whose goal is to order the stored data is an important feature of the VNB data base organization. The names of indicators are permanently stored and are displayed on the screen of a video display by means of identification numbers when queries are answered. The VNB specialists have made a special list of text names which do not occupy more than 40 characters on the screen, while the remaining 40 characters are allocated for numeric data. The data list is used to thoroughly capture the most important indicators, which include: The data entry date, the data output device (display screen or alpha-numeric printer), processing details (for instance, whether they involve computations), and the frequency with which the data are used.

The "permanent" part of the data base is established when the ABD begins operation; it includes the following records: "Type of transaction," "Bank," "Nation," "Currency," "Exchange rate" and "Term." The data base is filled in using magnetic disk devices.

Daily processing assumes the preliminary updating of the data base; information is written from a document onto a magnetic medium which is given an identification number, and is then entered into the computer. As the data are entered, they are rewritten to create a new file that includes data from fundamental and non-fundamental records.

After the data are entered into the computer, the ABD can begin functioning. One of its functions is to supply data necessary to make VNB management decisions in a question and answer mode. The bank directors compose uncoded, written queries. In order to facilitate filling out the queries, the Hungarian specialists have created a standard inquiry catalog; queries about data that do not require computations are distinguished from those that involve calculations (such as changes in the base and consecutive index distribution proportions).

The data base contains the following indicators for answering queries:

Indicators	Nation	Currency	Bank	Term
Credits	x	x	x	x
Deposits	x	x	x	x
% by deposit		x		x
% as credit		x		x
Credit balance			x	
Current transactions			x	

The type of transaction, the time it goes into effect, the bank and the currency must be indicated in the query to obtain the needed information. For example, the user requests the total debt owed to the "Chase-Manhattan" bank immediately after receiving short-term credit. In order to implement the query, "short-term credit," "Chase-Manhattan bank," "American dollars" must be indicated to implement the query. A special form is filled out, the operator encodes the query and enters it into the computer from a terminal. The answer is displayed on a video display screen or by an alpha-numeric printer.

The next function of the ABD is data storage and management. In order to implement this function, the Hungarian specialists have created a data base management system that controls the input of information (report function), the output of information (reference function), and reorganizes the data base.

We shall examine the operation of a data base maintenance system. Initially the user indicates a query, and the user program uses the query to access the data base. The management system analyzes the query and adds to it information that is stored in the data structure specification. Then the management system asks the operating system to initiate I/O (input in this case). The operating system searches for the data on the machine media, takes the data from the data base and places them into a memory buffer area (i.e., an area of memory intended for the data maintenance system). The data are then transferred to user memory space, and the data base management system issues the information to the user program. The data base maintenance system can be used interactively from remote terminals.

It should be mentioned that the ABD created by the Hungarian specialists makes minimal storage demands of the data processing system as a result of the carefully developed data base composition, data list, identifiers and query catalog. All of this has made it possible to centralize currency transactions, support interactive "man-machine" operation, and avoid losses resulting from processing delay when currency transactions are concluded.

The experience gained by the Bulgarian People's Bank in the creation of a data base is also of interest. The bank has developed a procedure for creating ABD in which five steps are distinguished: Data analysis, composition of class models, task analysis, and program design and development.

The data analysis task consists in recognizing relevant features of banking information, classifying it and ascertaining information sources. One of the

principal goals of this step is the design of flexible and efficient data for file processing. Initially the data are pre-analyzed and then modeled. The preliminary analysis includes determining the data analysis domain. In the modeling process the banking information is grouped and associations between independent data types are recognized, identifiers are created, the most important indicators are ascertained, and extraneous associations are examined. A class model is created as a result that comprises a logical structure of the data base and identifies subsystems and the associations between them. All parts of the economy that use the data and all banking information types are indicated in the model.

The principle tasks of each subsystem, the frequency with which they must be run, and their association with the class model are recognized at the next step (task analysis). The tasks are recognized and specified on the basis of a description of the operations accomplished in this subsystem.

It is important to note that one task may correspond to one or several interconnected operations. The class model access path is indicated for each of the tasks; for this one must determine the class that is essential to describe the other information classes in the sequence required for executing the banking functions. Then the task model is constructed.

The data base design stage includes the design of the logical and physical organization of the data base. The logical design includes specifying the data and the interconnection between them; the physical design solves data allocation problems with the goal of optimizing associations between the logical data.

The data base structure has been designed so that it need not be changed when the data processing system is reconfigured.

The data base management system is created when the physical design is complete. The Bulgarian People's Bank has developed two schemes for this purpose: The first is used when initially loading the data base, while the second is used when the data base is maintained during use.

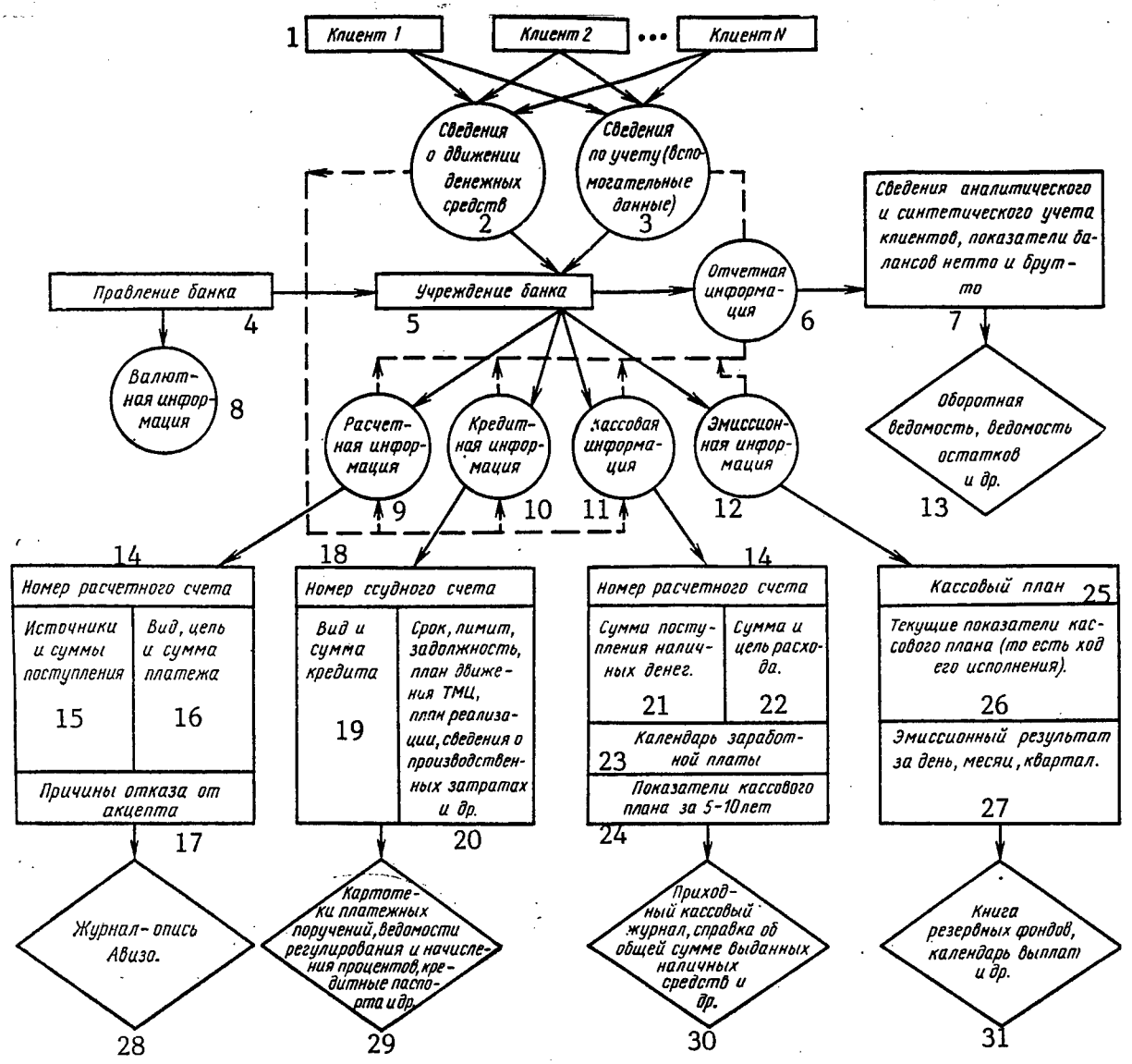
A favorable aspect of the Bulgarian People's Bank data base creation procedure should be noted: Information is organized not according to type (as at the Hungarian National Bank) but according to client, since the class model also includes users. This approach is more practically justified and accurate. The bank contacts its clients on a daily basis, and the data necessary to process and make decisions pertain to the activity of specific enterprises and organizations served by a given institution. This makes it possible to, firstly, take a systems approach to banking information processing, having correlated all of its different types, and secondly, to take into consideration one of the basic properties of banking information: semantic unity.

However, the latter is lacking in the class model developed by the Bulgarian specialists, i.e., it was intended solely for analytical and synthetic accounting. The creation of another, more efficient class model for the data bank based on this procedure and the experience of the Hungarian National Bank

is expedient, since the bank also provides crediting, cash and issuance services, drafts plans and supervises the production and economic activities of clients.

We note in connection with this that the following criteria are important for a USSR State Bank data base: Correct ascertainment of the nature of the banking information; segregation of the data into a principle data type classificationconsisting of rate, credit, currency circulation, financing, and class observance of state budget categories; information classification by temporal criteria (constant and variable); organization of all the information in terms of client groupings.

The class model can be represented in the following form (see diagram). After the date arrives at the bank, they are checked and written onto magnetic tape.



Data Base Structure.

(Key on following page.)

1--Client 1, Client 2, ..., Client N
2--Fund movement data
3--Auxiliary data
4--Bank administration
5--Bank establishment
6--Accounting information
7--Analytic and synthetic client accounting data, net and gross balance indicators
8--Currency information
9--Rate information
10--Credit information
11--Cash information
12--Issuance information
13--Balance record
14--Current account number
15--Receipt source and sum
16--Type, purpose and sum of payment
17--Causes of denied acceptance
18--Loan account number
19--Type and sum of credit
20--Term, limit, liabilities, commodity stock movement plan, implementation plan, production cost information, etc.
21--Cash receipt sum
22--Sum and purpose of expenditure
23--Wage calendar
24--Cash plan indicators for 5-10 years
25--Cash plan
26--Current cash plan indicators (i.e., current plan fulfillment)
27--Issuance result for day, month, quarter
28--Letter of advice register
29--Rate commission index, percentage adjustment and charge reports, etc.
30--Cash receipt register, information on total sum of cash withdrawals, etc.
31--Reserve asset book, payment calendar, etc.

This record will be comprised of four sets of data for each client: rate, credit, cash and accounting. The issuance area is filled in with summary indicators for all clients. The currency information is analyzed only at the level of the Government. The structure of each part is indicated by the diagram.

This data base structure makes it possible to obtain the requisite data about a client at any time, avoid executing a large number of laborious documents (letter of advice register, payment document index, credit passports, etc.). Thus the bank workers can work interactively with the data base, which makes it possible to conduct a more careful analysis and supervision of the production and economic activities of clients and to continuously monitor plan fulfillment.

COPYRIGHT: Izdatelstvo "Mashinostroyeniye," "Mekhanizatsiya i avtomatizatsiya proizvodstva," 1986

12678
CSO: 1863/19

UDC 621:65.011.56:005:658.532

AUTOMATION OF MATERIALS AND EQUIPMENT SUPPLY ACCOUNTING AND ANALYSIS DATA

Moscow MEKHANIZATSIYA I AVTOMATIZATSIYA PROIZVODSTVA in Russian No 4, Apr 86 pp 34-35

[Article by engineer and economist T. N. Malinovskaya]

[Text] The optimal and economic utilization of physical resources which are regularly supplied to sectors of the economy is taking on exceptionally great significance at the current stage of development of socialist society with the intensification of the economic system and accelerated scientific and technical progress.

The perfection of material resource management methods and the accounting and analysis of their use is to a large degree determined by the development of economic information processing facilities. The electronic computer has significantly improved materials and equipment supply (MTO) accounting and analysis, improved the speed with which data is processed and reduced the labor intensiveness of this operation, for the execution of which the YeS and small SM or the M5000 series of computer are used.

The most efficient MTO accounting and analysis systems presuppose the use of computer terminals. However, in practice standard physical asset accounting data processing systems frequently utilize punched cards, that are manually prepared from primary documents, as the data medium. Here the principle data conversion method is file processing. This processing method for accounting data, for example, is widely used at card centers that use the M5000 series computer (M5000 punchcard computer complex, and the M5010 and M5100 computer complexes), designed to replace punchcard computers. Batch processing is primarily used, which does not utilize even the limited multiprogramming data processing capabilities of the M5100 computer complex.

Equipping the lowest level computer facilities (small enterprises, rayon computer centers, etc.) with the M5000 electronic computers in place of punchcard computers has played an important part in the development of economic information computer processing systems, since it has made it possible to automate accounting data processing and enabled computer facility workers to assimilate new types of equipment and new technology.

The limited capabilities of the M5000 series computers currently do not

satisfy requirements for the continued improvement of MTO accounting and analysis data processing. One may assume that the production of the SM 1600 computer in place of the M5000 will have little effect due to its limited terminal device capabilities. The use of more powerful YeS computers for MTO data processing may be not be adequately efficient or simply impossible due to the absence of available computational resources, since this class of computer is utilized in most cases to process data for management of basic production processes. It is economically inexpedient to burden this class of computer with routine procedures to convert a large quantity of MTO accounting data in comparison with small computers used for this purpose.

Thus microprocessor-based mini- and microcomputers are finding ever greater use since they are simpler to operate and more accessible in terms of cost to medium-sized or even small enterprises. They can also be used for local data processing and in independent subdivisions of larger enterprises or organizations.

An integrated economic accounting system acquires special significance at the present time, when the efficiency of industrial enterprises must be increased not by adding resources but by realizing existing reserves and unused capabilities, and by the careful storage and optimal use of physical resources. In functional terms, the economic accounting of physical resources possesses two stages of inseparably interconnected processes. The first is the initial accounting stage which is comprised of calculation, economic operation commensuration and entering the obtained data into the primary documents. The second stage consists of the systematization and generalization of the initial accounting data, i.e., data accumulation, storage, retrieval and conversion with the purpose of subsequently generating the resulting indicators in a form suitable for use in enterprise management. Thus the second stage is also a data processing stage.

Data processing procedures were the first objectives of accounting mechanization and automation using various computer facilities. This was not the result of underestimating the significance of the initial accounting. The correct and timely documental representation of the actual availability and motion of physical assets is one of the most important requirements for monitoring their safe keeping and the opportune obtainment of trustworthy information required for the effective management of the production activity of enterprises and organizations. The difficulties of mechanizing and automating the procedures at the initial accounting stage consist in their complexity and labor intensiveness (the labor intensiveness of the initial physical asset accounting comprises about 55-60%) of the total accounting operation labor intensiveness). Therefore, the introduction of hardware has had little effect in this area, in which standard data processing methods are still used despite the great importance of initial accounting for obtaining objective indicators aimed at improving management. This situation frequently continues to occur even where systems that provide computer generation of summary initial accounting data function efficiently. The incompatibility between a highly efficient technology for producing accounting data consolidations and accounting forms that is based on third generation computers and standard initial data preparation methods is an essential factor that reduces the quality of the automated MTO management system's operation.

Lagging initial accounting automation was for a considerable time the result of an absence of efficient computer and organizational facilities oriented to the machine implementation of the procedures of this accounting stage. The continuous improvement of various data acquisition, recording, storage and transmission devices and peripheral equipment has made the automation of procedures at the initial accounting stage all the more necessary.

All procedures of both industrial enterprise MTO accounting and analysis stages can be implemented with specific hardware. However, different specialized data conversion points operate at the initial accounting stage in dependence on the hardware utilization. These points are subdivided into several types.

In the first type (which is most common at industrial enterprises), the initial data are initially recorded in a primary document, transferred to a machine-readable medium and then transmitted to a data processing center. Perforated tape or punched cards and magnetic media comprise the storage media, while the initial data preparation is centralized or decentralized. In the latter case it is accomplished at local data preparation points. The entire MTO accounting and analysis data processing cycle is quite a long process; the users have no direct connection with the computer. The results of all the solved tasks are outputed by printers and the report is transmitted to the user. The result of this is that many reports lose their strategic value and are not fully utilized since the urgency of the information has been lost and managerial decisions are made on the basis of approximate data, which adversely affects the quality of the decisions.

In the second type, the initial data preparation points record them on media during the machine compilation of primary documents. In most cases the initial data is pre-processed at this step with the corresponding computer facilities (accounting, invoicing and other machines), and the processing results recorded on the medium are transmitted to the computer center for further processing. This data preparation system can be very efficient if the process enables the medium with the initial data to be obtained when the primary document is compiled, which, unfortunately, is not always possible.

The third type of MTO initial data preparation system is, in our opinion, the most efficient and promising, and consists in creating specialized points equipped with small computers at storekeeping sites. This type of warehouse complex operates on an interactive basis; its hardware consists of a processor, a printer and specialized remote access points equipped with video consoles for automated work places (ARM) for storekeeping specialists. Usually there is a much greater number of staff members working in the same subdivision than terminals for them to use. This means that different subdivision specialists may take turns working at one terminal. The ARM provides the following functions: Display alpha-numeric data, set up and edit data, accept text and control data from the processor, transmit data from memory to the processor, transmit data entered at the keyboard to the processor while simultaneously accepting data from the processor, and transmit data from memory to the printer.

The application of a minicomputer-based ARM makes it possible to automate the preparation of MTO accounting and analysis data, since income and expenditure documents are generated while storing the corresponding quantitative changes in the computer's memory at the same time as reference information regarding the presence and motion of physical assets is obtained. With data preparation automation the document need not be used subsequently for entering data into the machine since it was created by the machine from this data. Considering the massive amount of primary MTO accounting and analysis documentation at the modern industrial enterprise, it is difficult to overestimate the value and efficiency of the automated generation of this documentation.

The use of the interactive mode in initial accounting automation means that the user can at any time request that any of the document forms automatically collected by the computer system be displayed on his terminal console, inspect them, and upon necessity, insert the required corrections into these documents before the document is printed out.

A corresponding search pattern is assigned to each primary document form stored in the data base when organizing interactive operation. The user formulates his query in natural language and the query is translated to a search prescript which is compared with the set of search patterns. That pattern which matches the search prescript is selected and the displayed search pattern is used to find the appropriate form for the documents to be filled out. The user can, if necessary, receive a hard copy of the primary document image on his screen by turning on the printer or obtain several copies of documents. The printer simultaneously specifies the entire communication protocol between the user and computer.

COPYRIGHT: Izdatelstvo "Mashinostroyeniye", "Mekhanizatsiya i avtomatizatsiya proizvodstva", 1986

12678
CSO: 1863/19

UDC 003.62:744.4:62

INTRODUCTION OF A UNIFIED GENERIC SYSTEM OF DESIGNATION OF PRODUCTS AND DESIGN DOCUMENTS

Moscow STANDARTY I KACHESTVO in Russian No 11, 1986 pp 47-48

[Article by N.S. Mishin, engineer, published under the rubric "Inter-industry Systems of Standards": "Introduction of a Unified Generic System of Designation of Products and Design Documents"]

[Text] Reducing the nomenclature of assembly units and components in an equipment being designed is a desirable goal, which at first glance is a truism hardly worth discussing. This is true, however, only at first glance. While people in industry and operators of equipment are best aware of the need for this reduction, regrettably, the designers underestimate it. What is the cause of this?

A study of the existing automatic rotor lines developed before 1970 show that they are saturated by unjustified diversity of assembly units and components and that this is due to various reasons--mainly, an object-based system of designation of products and design documentation.

With this system all components of a line are designated within the notation system of the line itself. This means that each component and assemlby unit contains in its code a designation (brand) of the automatic production line where it is used.

The code of a production line consists of letters and numbers, e.g., LGSh-400, LGSh-12 and LGSh-V/600. Letters and numbers express various notions: the names of industrial operations performed, the names of the workpieces, the number of workpieces processed simultaneously, the magnitude of the processing force, the productivity, etc.

For example, in the above three production lines, symbols in the notations have the meaning illustrated by the following table.

This example shows that the codes of the lines contain different numbers of symbols, i.e., the code lengths are different; the same symbols may designate different notions and, conversely, different symbols may refer to the same notions. This creates difficulties for computer processing of information contained in design documents.

Line notation	L	G	Sh	V	400	12	600
LGSh-400	Line	Hot	Stamping	-	Force in tons	-	-
LGSh-12	Line	Nuts	Hexagonal	-	-	Nut diameter, mm	-
LGSh-V/600	Line	Sockets	Stamping	Socket type	-	-	Output

In addition to the line brand, the designation of a design document describing a component or assembly unit contains its designation within the line where it is used.

The line is divided into groups: The first assembly unit of the first group is denoted by 1-1, the second by 1-2, the third by 1-3; the first assembly unit of the second group is denoted by 2-1, the second by 2-2, the third by 2-3, etc. The first component of the first assembly unit of the first group is denoted by 1-101, the second by 1-102, the third by 1-103; the first component of the second assembly unit of the second group is denoted by 2-101, the second by 2-102, the third by 2-103, etc.

This system is responsible for the fact that in different lines the assembly units and components that are different can have the same designations, which makes it difficult to conduct a rapid search of the existing design solutions or analogs so as to borrow these concepts in new designs or to group components according to their design similarities.

A generic classification system for denoting products and design documents makes it possible to designate each product and the document describing it on the basis of a classifier; such a classifier is developed in advance regardless of the use of a particular assembly unit or component in any specific production line.

As a basis for the generic classificatory notation, the developers of automatic rotor lines chose the draft YeSKD [Unified System of Design Documentation] Classifier (1970 edition). After using this classifier for 15 years some conclusions can be drawn.

Of special interest to designers of automatic rotor lines in YeSKD Classifier are the following classes:

assembly units of a general engineering type--class 30* (footnote) (the class numbers are given as they are indicated in the latest edition of ESKD Classifier);

components of the "body of revolution" type--classes 71 and 72; and

components of the "not body of revolution" type--classes 73 and 74.

The main principle of component classification--according to the geometric shape--makes the ESKD Classifier an efficient tool for maximum unification of components and organizing group-based methods of processing them.

A hierarchical method of classification with five levels provides a consistent increase in the specificity of the features represented on drawings of products at different degrees of classification division. Each next place in the code of a classification characteristic describes the component at the next step of specificity as compared with the preceding digit.

The generic system of notations based on the YeSKD Classifier does not associate the components and assembly units with any particular product and makes it possible to find existing solutions and analogs which can be borrowed in the development of new concepts; this makes it possible to create engineering, design and industrial process documents and introduce group-based methods of machining; the operation, planning and scheduling can also be done more efficiently with the aid of a computer. The advantages of the generic system over the object-tied system are obvious.

The introduction of the generic system of notations presupposes a series of organizational and technical steps; above all, the following must be done: The system of the YeSKD Classifier must be studied closely by the broad groups of specialists at the enterprise (designers, process engineers, production engineers, standardization officers, designers of CAD systems and ASUP [plant management automation systems], etc.). The introduction of the system should involve also the following measures:

creating and installing an automated or mechanized system of search of design solutions and their borrowing analogs;

formation of specifications, component certificates and other service documents on a computer; and

developing a system of current planning and production scheduling with the aid of a computer, etc.

The standardization service should be the main agent in studying and spreading the use of the generic system. This is a difficult task. The difficulties stems, above all, from the fact that until now there have been no easily understandable manuals presenting the capabilities of the generic system; there is practically no information about the experience of introducing and maintaining the system at the leading enterprises across the nation.

While this system is still being studied and prepared for introduction, psychological difficulties have to be confronted, because the generic system is less obvious to human perception than the object-based system (it is mainly intended for computerization). In addition, the generic system constrains the designer, who is forced to make the maximum possible use of existing and practically tested designs of assembly units and components when creating a new product, which is by far not always a welcome change for

a designer accustomed to complete freedom of action provided by the object-tied system of notations.

Practical experience suggests, however, that at an enterprise where the system of mechanized or automated search of design documents has been installed even on an experimental basis the time for familiarizing oneself with such a system will be greatly reduced thanks to the dramatic results in raising the productivity of the designers and the substantial reductions of costs in product manufacturing.

With a generic system it is possible at a minimal labor cost to prepare systems classifying geometrically similar and relatively simple components and to compile picture catalogues for classification of more complex components and assembly units. With these tables and catalogues the search for analogs of assembly units to be borrowed in new designs becomes much easier; it also helps to resolve various problems in product standardization.

The next step in developing a system of analog search at our enterprise was the introduction of a system of mechanized document processing on a microfilm basis [SMODM]. The system, although not without snags, can be used to find design and industrial process documents.

COPYRIGHT: Izdatelstvo standartov, 1986

9922
CSO: 1863/137

UDC 681.3:658.511.011.56

ON PROCEDURE AND RULES FOR PRESENTATION AND HANDLING OF DESIGN DATA ON MAGNETIC MEDIA WITHIN YeSKD FRAMEWORK

Moscow STANDARTY I KACHESTVO in Russian No 9, Sep 86 pp 20-22

[Article by Candidates of Technical Sciences Yu.B. Fratkin, A.S. Tretyakov and I.M. Yakubovich, and K.V. Novikova]

[Text] The current stage of scientific and technical revolution is related to widespread implementation and practical application of computer technology in various spheres of national economy. One of the mainstream routes for intensification of industrial production is replacement of physical, routine and low-productive labor and widespread utilization of its intellectual forms, based on rich capabilities of computers. The main role is to be played by flexible manufacturing systems (GPS), wherein the most modern technological equipment, computerized control equipment and robotic systems, automated design systems (SAPR), automated systems for production engineering (ASTPP) etc. are used.

At the same time, uncoordinated application of various automated systems will not result in significant acceleration of production engineering, because in this case a considerable amount of "manual" labor is required in order to prepare or transform and then code the input information. Only combination of individual automated systems, designed for various purposes, into a unified integrated system, development of software-information interfaces and a common data bank can bring in a qualitative leap in labor productivity of various categories of employees, involved in design and organization of production and manufacturing.

Of the great number of problems, related to development of integrated systems, the most critical one is the problem of informational interaction between various automated systems, in the first place, between SAPR and ASTPP.

One should take into account that automation of production and control of products under development is based on a database which is formed at the automated design stage. At the same time, in order to manage a production process much more comprehensive information is required than that obtained from an SAPR.

As a rule, SAPR data are at present entered into an ASTPP database manually. In order to eliminate time losses for interoperation movement of information from an SAPR to an ASTPP and, if necessary, in the opposite direction, it is

necessary to develop a unified database that contains all necessary information for supporting design and production engineering processes. However, such method for data exchange between various automated systems is only applicable to systems that use uniform hardware.

For systems operating on different hardware, informational interaction can be accomplished by transmitting information files on magnetic media or by transmitting information via communication channels. Under present conditions, the first method seems to be the simplest, as far as its practical implementation is concerned.

As such, statement of the problem of document presentation on a magnetic medium is not new nowadays. But with existing approaches to solving this problem (within the framework of standard technical documents that are in effect), one should provide, as the starting position, identity of informational contents of a document both on paper and magnetic media, i.e. documents should only differ in the form information is presented on a medium. With this approach, information contained in a set of design documents is distributed on a magnetic medium in strict correspondence to its belonging to a specific type of documents. We shall conditionally call it a "distributed form of recording."

Unlike the distributed form, we shall call the method we are proposing for presentation of design results an "integrated form of recording." According to this method, information is not linked to specific type of documents. And this means that data in a file on a magnetic medium can be used not only for producing documents but also for other production problems, for instance, for solving production engineering problems. In addition, this form of recording eliminates duplication and simplifies automated entering of engineering changes. This is the main advantage of the integrated form of recording documents, and implementation of this form is one of the elements of an automated "design - production" system.

A magnetic medium containing a product design data file can be delivered to a manufacturing plant. This paperless type of relations between the developer and the manufacturer can only be realized after solving a number of psychological, technical and organizational problems.

Among these problems are:

1. Determining the status of the design file on a magnetic medium from the standpoint of relations between the developer and the manufacturer.

2. Establishing a procedure for delivering a data file on a magnetic medium in the integrated form of recording, which calls for a principally new approach.

3. Determining procedures that regulate relations between the developer and the manufacturer in the process of delivering data on a magnetic medium.

4. Establishing rules for designation of a new type of a document.

5. Determining and regulating interdependence between a file on a magnetic medium in the integrated form of recording and KD [design documentation] in a traditional form. The need and conditions for combined application thereof.

6. Determining officials that approve and sign the data file on a magnetic medium, and the contents of information required for its application and handling and the place for storing this information, i.e. determining what is traditionally called a "Title Block."

7. Establishing rules and procedures for storage, accounting for and maintenance of the data file on a magnetic medium in the integrated form of recording.

8. Establishing rules and procedures for entering engineering changes.

Solution of the above problems, that takes into account their reduction to specific rules and requirements, calls for rather deep studies. Therefore, within the framework of this article we shall only dwell on directions for solving these problems.

1. The problem with the status of a design file on a magnetic medium is that the file cannot be linked to a specific document type and that information contained in the file can be used for a broader range of problems.

It is important to solve this problem because it is a foundation for solving the rest of the above listed problems.

The status of a design file on a magnetic medium can be established, based on its main purpose and contents.

The contents of a file is determined by its name. The file contains data on the product, its design and components, in other words, the same data that are contained in traditional design documents. But in our case, these data are not duplicated and are collected in the file in a predetermined structure.

The main purpose of data in the file is to be used for manufacturing products, i.e. the main purpose of the file is similar to the function of design documents.

According to an encyclopedia definition, the concept of a "document" also covers a magnetic tape with information, recorded on it and intended for transmission in time and space. It follows from the above, that the status of a design file recorded on a magnetic medium in the integrated form is referring the file to the concept of a "document" in the design documentation sphere.

This document comes into legal force after the rest of the above listed problems have been solved in accordance with GOST 6.10.4-84 "USD [Unified System of Documentation]. Giving Legal Force to Computer Created Documents on Machine Medium and Machinegram. Principal Provisions."

2. In the case of delivering a document on a magnetic medium, problems must be solved that are not only related to delivery processes as such, but also to preparation and presentation of documentation to be delivered.

The essence of the problem is in solving principally new problems, different from traditional forms of delivering a set of design documents. For instance, can a document on a magnetic medium be delivered separately or only as a part of a complete set of product documentation? Should the delivery be allowed at earlier stages of product development, prior to delivering a complete set of documents, with the purpose of using data, recorded on the magnetic medium, for solving production engineering problems? How should the authenticity of data, recorded on a magnetic medium, be confirmed at the moment of delivery of the medium? What kind of documents should accompany the file and how necessary they are in the delivery process? Should one differentiate the procedure for delivering a document on a magnetic medium depending on what is being delivered, the original, a duplicate or a copy?

Solving these problems will make it possible to formulate concrete rules for delivering documents which are recorded on a magnetic medium in the integrated form. These problems should be regarded as principal directions in solving the delivery problem.

3. The problem of delivering a document on a magnetic medium is part and parcel of the problem of establishing relations between the developer and the manufacturer.

The main problems in disclosing and solving this problem are reduced to:

delineation of responsibilities of the developer that delivers a magnetic medium with a data file;

delineation of rights of the manufacturer that accepts the document on a magnetic medium;

delineation of rights and responsibilities between the developer and the manufacturer, depending on whether the original, a duplicate or a copy of the document on a magnetic medium is delivered.

The aceptability of delivering the data file on a magnetic medium should be determined by mutual agreement of the parties and specified in the TZ [design assignment] for product development.

In our opinion, the basic approach to solving this problem is to get the manufacturer to participate in all stages of automated design of the product. For his part, the developer must ensure obtaining of zero-defects results of automated design and zero-defects recording thereof on a magnetic medium. By mutual agreement, complete or limited manufacturer's control of the delivered data file, using visual means for data representation, can be established.

4. Because the problem in question is a document with the integrated form of recording data on a magnetic medium, solution of the problem of identifying such document within the design documentation set has the following aspects:

on one hand, regarding a data file on a magnetic medium as a design document, we have a formal right to assign the document the decimal designation of the product and add a letter code, identifying this document type;

on the other hand, correspondence between data on a magnetic medium and various types of design documents and the capability to obtain these documents from the medium with their own designations does not allow a formal approach to assignment of the designation. In the case under consideration, one does not have to look for analogs with rules for designation of design documents on punched media which are established in State standards. It is well known that these rules provide for designation of a document on a punched medium, identical to designation of the type of a "traditional" document, obtained from this punched medium.

Apparently, the solution should be derived, based on the very contents of the proposed by us document on a magnetic medium. Indeed, this document contains a data file that can be regarded from the design documentation standpoint as a set of data,that are related to various types of design documents and provide the capability to obtain the documents in a visual form. In other words, a document on a magnetic medium corresponds to a set of design documents for a certain entity. And this calls for determination of a letter code, identifying the form and the complete set of the document under consideration. This code should be added to the decimal designation of the product or the subassembly.

5. A design data file on a magnetic medium can include all information on the product, required for manufacturing thereof. This information will correspond to the complete set of design documents.

The need for obtaining design documents in their traditional form is determined by production, by its technical equipment and its readiness to apply results of design on a magnetic medium to the manufacturing process.

Nowadays, a manufacturer cannot do without traditional design documents, which are needed for controlling product manufacturing, visual entering of engineering changes and use during manufacturing. And this means that along with the document on a magnetic medium, one should also deliver a portion of traditional design documents. There should be a certain relationship between these documents, specified in the finished product data list. Such a relationship has already been implemented for documents on punched media, recorded in the list of machine media. The latter is in turn recorded in the "Documentation" section of the product data list.

This solution should also be preserved for a document on a magnetic medium with the goal to record this document in the common list, together with documents on punched media.

6. Among the main problems in specifying requirements to the "Title Block" of a document in the integrated form of recording on a magnetic medium are:

specifying officials, who are responsible for information, stored on the media, and who sign the document;

establishing reasonable contents of the "Title Block" columns, taking into account the feasibility of obtaining more detailed information on data on the magnetic medium directly during data printout;

presenting in the title block information on correspondence between data on the magnetic medium and concrete types of design documents.

Solution of other problems, related to the title block structure, contents and location, follows from the solution of the above listed problems.

It is most feasible to base the definition of the original, a duplicate and a copy on definitions, established by GOST 6.10.2-84. In this standard, regarding a document on a magnetic medium as the original, a duplicate or a copy is related to the time sequence of producing it.

7. Accounting for, storage and handling of documents on a magnetic medium are established separately for the original, a duplicate or copies. One should not link solution of these problems to the search for principally new directions.

Basic provisions for accounting for, storage and handling of documents should be based on appropriate rules and provisions of YeSKD [Unified System of Design Documentation] standards. The differences will only be in storage and handling of a document on a magnetic medium, which is related to providing appropriate environment for preserving physical properties of the magnetic medium, as well as to concurrent presence of "traditional" design documents and the document on a magnetic medium they were produced from.

8. According to GOST 6.10.4-84, the original of a document on a magnetic medium can only be changed by the organization that created the document.

This requirement does not provide for delivery to the manufacturer of the appropriate set of software and hardware that give the manufacturer broad opportunities for application of documents on magnetic media. Besides, this requirement does not match the basic requirement of YeSKD standards on making engineering changes in KD, according to which any enterprise has the right to make changes in originals or copies, depending on what document type it is the holder of. Only the holder of the original has the right to issue an engineering change notice.

The document on a magnetic medium, examined in this paper, is referred to design documentation; therefore, when making changes in it, it is logical to follow the rules, presented in YeSKD standards. Whether the manufacturer is given copies or originals of the design data file on a magnetic medium, he should be provided with means that facilitate the process of making changes in the file. To this end, manufacturers should be provided with a common set of unified software and hardware that make it possible to perform all necessary operations with the design data file on a magnetic medium (producing traditional documents, making changes and solving other problems).

The developer still has the right and responsibility to issue an engineering change notice, if he is still the holder of the origial.

This approach to solving the problem of entering changes makes it possible to enter them in a timely manner, to quickly react to an engineering change in the case of automated solving of production engineering problems and more effectively use computer technology in production.

Presentation and issuance of engineering change notices should be performed in accordance with GOST 2.505-82.

Solving the above listed problems and compiling them in concrete standard technical documents will make it possible for enterprises to speed up application of design results in production and create prerequisites for accelerating development and industrial implementation of automated systems, that eliminate "paper" relationships between the developer and the manufacturer of a product and support an automated "development-manufacturing" cycle.

COPYRIGHT: Izdatelstvo standartov, 1986

12770
CSO: 1863/66

EXAMINATION OF NEW PROCEDURE FOR DEVELOPMENT AND COORDINATION OF TECHNICAL DOCUMENTATION AT REGIONAL MEETINGS

Moscow STANDARTY I KACHESTVO in Russian No 9, Sep 86 pp 69-71

[Text] As a measure for implementation of the USSR Council of Ministers resolution of 01.14.86 No 65 "On Improving The Procedure for Development and Coordination of Technical Documentation During Development and Production Start-Up of New (Modernized) Products in Machine Building Industry," Gosstandart [State Committee for Standards] had scheduled a series of meetings with representatives of enterprises and territorial offices of Gosstandart in main industrial regions of the country in order to clarify the new procedure as it applies to specific production conditions and requirements of NTD [standard technical documents].

The first such meeting for Baltic republics and Belorussia was organized by the Lithuanian republican administration of Gosstandart. The meeting took place on May 21-22 in the city of Panevezhis. Over 300 people took part in it, including 200 people invited from Latvia, Belorussia and Estonia.

The new procedure calls for significant changes in standards of the State Standardization System (GSS), the System for Product Development and Production Start-Up (SRPP), and the Unified System of Design Documentation (YeSKD), dealing with certification, testing and state supervision. This is why leading professionals of VNIIS [All-Union Scientific Research Institute of Standardization] in all of the above areas took part in the meeting. A secretary of Panevezhis city Party committee A.P. Ryabov opened the meeting.

Department manager of VNIIS, Candidate of Technical Sciences O.V. Yaremenko presented a report "New Procedure for Development and Coordination of Technical Documentation in Product Development." He presented basic principles of changes in the procedure for development of new (modernized) products that follow from the resolution. In particular, he noted that decentralization of the process of developing documentation for new products is being contemplated, that the developer and the customer are given more rights, and that relations between the two parties must be built on a contractual basis, which will increase their mutual responsibilities.

In order to eliminate procedural provisions from the SRPP, criteria for requirements which should be regarded as mandatory were developed. Based on this, GOST 15.101-80 and GOST 15.304-80 were converted to procedural instructions and a number of items in GOST 15.001-73 were eliminated,

including converting appendices 1 and 2 from mandatory to recommended.

He then spoke in great detail about the new procedure for development, coordination and approval of the design assignment (TZ), delineation of responsibilities of the customer and the developer in developing new products, and on simplifying the procedure for developing a TZ for individual production. The report was illustrated with examples from the practical work with industry.

Head of a sector of VNIIS, Candidate of Technical Sciences I.V. Pryazhinskiy presented a report "On Principal Provisions of New Version of GSS and Specific Features of Development, Coordination, Approval and Recording of Specifications [TU]." He reported on the main changes made in the GSS. The new version of the set of GSS standards is aimed at the development of a system of NTD and at conducting a unified state technical policy at all levels of management of national economy with the following objectives:

develop and manufacture products with parameters that match or exceed the highest world level;

optimize the structure and contents of the NTD fund in order to improve its dynamic character in the process of integrated solution of problems of accelerating the NTP [scientific and technical progress], and improving the technical level and quality of products;

simplify the procedure for development, coordination and approval of NTD, increase the responsibility of Ministries and agencies for timely preparation and practical application of NTD.

He listed 12 state standards that comprise the new GSS version and will be made effcetive as of 01.01.87.

Rules and regulations regarding organization and performance of work on standardization are coordinated with the activity on planning economical and social development of the USSR, on organization and performance of NIR [scientific research work] and OKR [experimental design work], on development, production start-up, manufacturing and consumption (operation) of products, on pricing, on utilization of inventions and discoveries etc.

I.V. Pryazhinskiy also reported on principal provisions of the new GOST 1.3-85 "GSS. Procedure for Coordination, Approval and State Recording of Specifications." They are reduced to the following:

the composition of TU and the contents of sections thereof is determined by the developer in accordance with specific features of products;

TU must be coordinated at the acceptance commission meeting with the customer, the union and state supervision agencies, in accordance with statutes thereof. If a product is implemented in production without being put through the process of acceptance by the acceptance commission, then TU must be coordinated with these organizations by simultaneously sending them copies of TU;

TU are approved by head (leading) Ministries by types of products, or else the latter relegate this right to Ministries that are not head (leading) ones by the product type;

TU for new (modernized) products are approved in accordance with a procedure established by the developing Ministry;

TU for consumer goods, made from production scrap, are approved by enterprises of republican Ministries and consumer cooperative societies;

TU must be recorded at territorial (republican, kray) offices of Gosstandart according to location of the developing organization.

At the same time, the speaker noted that since the beginning of the year the recording procedure has been repeatedly corrected by Gosstandart letters. This condition badly hinders organization of work on standardization in industry and impedes activities of state supervision agencies.

Senior Scientific Associate of VNIIS R.Kh. Sulpovar presented a report "Compilation of Technical Level and Product Quality Card [KU] and Application Thereof in Product Certification."

In his report, R.Kh. Sulpovar presented basic provisions of the renewed GOST 2.116-84 "Technical Level and Product Quality Card" and first results of industrial implementation thereof. An important in principle provision of the standard is that it specifies the list of products for which a level card must be developed.

The standard establishes a new procedure for development and maintenance of a level card and new requirements to the contents of information in it, and delineates responsibility for completeness and truthfulness of information, presented by the leading (head) organization for this type of product, the developing organization, the customer organization (principal customer) and the manufacturing enterprise.

In order to develop a common informational basis for decision making and coordination of these decisions at all management levels over the entire product life, the standard regulates requirements to selection of nomenclature of quality indices of products under evaluation, to establishing a reference model, and to the contents of information data on analogs and the product model being replaced.

The speaker then told the audience about the role a KU plays in product certification and the procedure for selecting the reference model at the full-scale production stage.

In conclusion he noted that implementation of GOST 2.116-84 had demonstrated that the work, aimed at organizing systems for acquisition and processing of information using the best domestic and foreign analogs and at developing methods for evaluation of the technical level and quality of individual types

of products should be seriously improved. This conclusion was corroborated by a large number of questions, dealing mainly with hardness in obtaining truthful information on the best up-to-date models.

Head of sector of VNIIS M.I. Beylina presented a report on the procedure for product certification and analysis of certification results. She noted that at present a large part of manufactured products are not covered by certification. Particularly, among these are products that define the profile of Ministries (agencies) and manufacturing enterprises.

In order to overcome this condition, a decision has been made to develop, in coordination with Gosstandart, All-Union and Union-republican USSR Ministries (agencies) and with Counsils of Ministers or on their behalf by Gosplans of Union republics, lists of products not subject to certification by quality categories.

According to a Gosstandart decision, in performing this work, one can, beginning in March, 1986, add the following products to the previously established nomenclature of products not subject to certification:

raw materials, substances that underwent industrial processing but are used as intermediate substances in technological processes of manufacturing finished products;

semi-finished products for intra- and interindustrial application (castings, forgings, stampings, welded steel structures etc.), for which only application indices are specified in NTD;

products of machine building industry, for which only application indices are specified, that are used as components of finished products;

cultural and consumer goods, for which one does not have to specify in NTD indices of reliability, material content and energy consumption during operation. As before, the main shortcoming of certification is groundless attribution of products to quality categories.

Examination of materials of state certification commissions (GAK), conducted by VNIIS, has demonstrated that more than 17% of decisions, submitted since 01.01.86, turned out to be groundless. The most typical cases of rejecting certification or down-rating a product to a lower category are:

absence of comparison with an analog and of other data that could justify a GAK certification decision;

erroneous comparison of technical and economical indices of the certified product and the selected analog;

discrepancy between specifications and state standards;

violation of GSS requirements during NTD development and confirmation;

violation of established requirements while conducting tests etc.

The report also covered problems that were of interest to meeting participants, such as coordination of certification plans with Gosstandart, coordination of durations of NTD and quality categories, principal provisions on GAK rights and responsibilities for certification results.

Senior Scientific Associate of VNIIS, Candidate of Technical Sciences M.G. Dolinskaya presented a report on "Testing As Basis for Objective Evaluation of Product Quality During Development and Certification Thereof."

The need for objective evaluation of the technical level and quality of products during development and certification thereof, introduction of certification of products of machine building industry, increased volume of exported products and hence the need for mutual acceptance of test results by product suppliers and customers, both inside the country and abroad, called for serious improvement of testing in the country.

Procedural guidance and control of correct conductance of state testing of the most important types of products of industrial, technical, cultural and consumer application is performed by Gosstandart. Together with Ministries and agencies, it has approved by now 185 head organizations on state testing of the most important types of products (GOGIP) and developed about 30 NTD, regulating activities thereof. The GOGIP network is created in order to accumulate experience of state testing and to optimize methodology thereof, and in the first place to determine conditions for obtaining objective truthful results.

As was noted in the report, objectivity and truthfulness of test results are ensured if the following conditions are observed:

the tests must be conducted according to test programs and procedures, developed and coordinated according to established procedures, using certified test equipment and calibrated instrumentation, by certified testing departments;

indices and accuracy standards of test instrumentation, reproducibility of test conditions, methods for processing test data, the form of test results presentation, test control plans, control standards and decision rules, specified in test programs and procedures, must be unified and meet established requirements;

certification of test organizations and departments must state that expertise of personnel and technical base for tests ensure that tests are conducted in full compliance with NTD specifications.

Then M.G. Dolinskaya set forth principal provisions of the authors' version of a draft of procedural instructions "Test Program and Procedure. Basic Requirements," that is being developed by VNIIS.

In developing this document, one took into account provisions of resolution No 65, regarding test programs and procedures. Attention was mainly paid to certification test programs and procedures and to conditions, under which

results of other types of tests may be used for certification.

Head of department of VNIIS, Candidate of Technical Sciences Ye.A. Avsiyevich presented a report "State Supervision of Compliance With Standards During Product Development."

He set forth the essence of a new methodological approach to organization and performance of state supervision of compliance with standards during product development. Attention of professionals of territorial offices of Gosstandart was mainly drawn to the need to concentrate, when auditing developing organizations, on end results rather than on procedural problems and control of standards.

Procedural recommendations on checking compliance of the technical level of newly developed and modernized products with the requirements of GOST 2.116-84, SRPP and OTT [not further identified] standards for groups of similar products were given.

The speaker also examined problems of further improvement of scientific and technical aspects of state supervision of standards and quality of products.

All reports were listened to with great interest and generated a lot of questions. Some of the questions have to be answered at Gosstandart level; therefore, meeting participants addressed the following proposals to Gosstandart:

1. For more efficient organization of work on industrial standardization, reduce the number of letters mailed out by Gosstandart that include additional requirements for active standards and other NTD. Handle all necessary clarifications and additions as changes, in accordance with established procedures.

2. Taking into account the need to develop products that match or exceed the highest world achievements, incorporate in the plan the development of procedural documents on determination and evaluation of competitiveness and the procedure for comparison with the best world models.

3. Solve the problem of the procedure for recording a level card in the VTsIO [not further identified] and in territorial offices of Gosstandart, if they are located in different territorial zones.

4. Accelerate development of an SRPP standard for consumer goods.

5. Clarify the certification procedure, taking into account all corrections, introduced by Gosstandart letters, including those dealing with lists of products not subject to certification, due dates and procedures for recording of GAK decisions etc.

6. In order to establish standard due dates for product renewal, provide in appropriate standards clear definitions of the following concepts: new products, newly mastered products, modernized products, the simplest products, product development, product modernization.

COPYRIGHT: Izdatelstvo standartov, 1986

12770
CSO: 1863/66

UDC 681.32.069:681.2.03

MIXED LOGIC AND HARDWARE MODELING OF DIGITAL-ANALOG APPARATUS

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86
(manuscript received 14 Dec 84) pp 29-34

[Article by A. I. Petrenko, G. D. Kiselev and V. Sh. Giorgizova-Gay]

[Abstract] One trend in the development of automatic electronic hardware design systems is the creation of program systems for multilevel modeling of complex objects, allowing representation of individual parts of the electronic device being modeled by models of widely varying complexity, decreasing the influence of shortcomings of any given level on the overall results of modeling. Mixed logic and hardware analysis systems combine the accuracy of hardware modeling for critical parts with the speed and economy of software logic modeling. This allows analysis of circuits of practically any organization, digital, analog or mixed; use of a mathematical model with an arbitrary number of digital and analog components; modeling only of components for which the signals at the inputs have changed their logic value; and simultaneous modeling of several analog components with different integration step for the mathematical model of each one. The organization of a logic-electrical modeling system is diagramed and described. Figures 5, references 11: 6 Russian, 5 Western.

6508/5915
CSO: 1863/139

BOOK ON EVOLUTION OF AUTO-WAVE VORTICES PUBLISHED

Moscow NOVOYE V ZHIZNI, NAUKE, TEKHNIKE: SERIYA MATEMATIKA KIBERNETIKA (EVOLYUTSIYA AVTOVOLNOVYKH VIKHREY) in Russian No 8, 1986 (signed to press 23 Jun 86), pp 1, 2, 48

[Annotation, table of contents and bibliography from the book "The Mathematics of Cybernetics: The Evolution of Auto-Wave Vortices (heart waves)" by Valentin Izrailevich Krinskiy, Doctor of Physical and Mathematical Sciences, Professor, Lenin prize laureate and manager of an autowave processes laboratory; Aleksandr Berelyevich Medvinskiy, Candidate of Physical and Mathematical Sciences, scientific worker, expert in the area of electrophysiology and stimulated myocardium waves; Aleksandr Viktorovich Panfilov, Candidate of Physical and Mathematical Sciences. scientific worker and expert in the area of autowave process modeling; reviewed by S. P. Kurdyumov, corresponding member of the USSR Academy of Sciences, 1630 copies, 48 pages.]

ANNOTATION

This brochure is devoted to the theory of spiral waves, the theory and significance of which is very important today. This brochure describes waves in the heart, the eye retina, and the core of the brain. The analysis of these phenomena helps in diagnosing illnesses, outlines new ways for constructing computers and so forth. It became possible to solve these tasks through the use of computers and mathematical modeling methods in electrophysiological research.

This brochure has been published for lecturers, students and instructors in national universities.

TABLE OF CONTENTS

CHAPTER 1. Introduction 3
CHAPTER 2. Vortices in very simple mathematical models of active materials 5
CHAPTER 3. Reverberations, whorls, discharges and twists 9
3.1. Mathematical models for simulation generation and propagation . -
3.2. Spiral waves 15
3.3. Three-dimensional autowave sources 20
CHAPTER 4. The evolution of myocardium autowave vortices 25

4.1. Heart design and operation 26
4.2. Experimental research into myocardium vortex structures . . . 30
4.3. Computer reconstruction equipment for wave images in the heart 38
CHAPTER 5. Managing autowave vortex evolution 42
CHAPTER 6. Conclusions 46
BIBLIOGRAPHY 48

BIBLIOGRAPHY

Akhromeyeva T. S., Kurdyumov S. P., Malenetskiy G. G., "Paradoksy Mira Nestatsionarnykh Struktur" [Paradoxes in the World of Non-stationary Structures], Moscow, Znaniye, 1985.

Balkarey Yu. I., Golik L. L., Yelinson M. N., "Avtovolnovye Sredy (ispolzovaniye v elektronike)" [Autowave Mediums (their use in electronics)], Moscow, Znaniya, 1985.

Ivanitskiy G. R., "Biologicheskiye Mikroustroystva" [Biological Microsystems], handbook, USSR Academy of Sciences, 1984, No 3.

Ivanitskiy G. R., Krinskiy V. I., Selkov Ye. Ye., "Matematicheskaya Biofizika Kletki [Mathematical Framework Biophysics], Moscow, Nauka, 1978.

Kadomtsev B. B., Rydnik V. I., "Volny Vokrug Nas" [The Waves Around Us], Moscow, Znaniye, 1981.

Krinskiy V. I., "Fibrillyatsiya v Vozbudimykh Sredakh [Filbrillation in Excited Mediums], in the collection "Problemy Kibernetiki" [Problems in Cybernetics], 1968, No 20.

Krinskiy V. I., Mikhaylov A. S., "Avtovolny" [Autowaves], Moscow, Znaniye, 1984.

Kurdyumov S. P., Malenetskiy G. G., "Sinergetika -- Teoriya Samoorganizatsii. Idei, Metody, Perspektivy" [Synergetics -- The Theory of Self-Organizing. Ideas, Methods, Futures], Moscow, Znaniye, 1983.

Mazur N. A., "Vnezapnaya Smert Pri Ishemicheskoy Bolezni Serdtsa" [Sudden Death with Ischemia of the Heart], in the book "Rukovodstvo po Kardiologii" [Managing With Cardiology]. T. Z. Moscow, 1982.

Nakolis G., Prigozhin I., "Samoorganizatsiya v Neravnovesnykh Sistemakh" [Self-Organizing in Nonequilibrium Systems], Moscow, MIR, 1979.

Rambidi N. G., Zamalin V. M., "Molekulyarnaya Mikroelektronika: Istoki i Nadezhdy" [Molecular Microelectronics: Sources and Expectations], Moscow, Znaniya, 1985.

Rozenshtraukh L. V., "Rasprostraneniye Vozbuzhdeniya po Razlichnym Strukturam Serdtsa" [Simulation Distribution in Various Heart Structures], in the book "Fiziologiya Krovoobrashcheniya. Fiziologiya Serdtsa" [The Physiology of Circulation. The Physiology of the Heart], Leningrad, Nauka, 1980.

COPYRIGHT: Izdatelstvo "Znaniye", 1986

12511
CSO: 1863/107

UDC 631.55:001.18

SIMULATION MODELS AND METHODS OF MAKING DECISIONS IN PROGRAMMING HARVESTS

Moscow DOKLADY VSESOYUZNOGO ORDENA LENINA I ORDENA TRUDOVOGO KRASNOGO ZNAMENI AKADEMII SELSKOKHOZYAYSTVENNYKH NAUK IMENI V. I. LENINA in Russian No 2, Feb 86 (manuscript received 17 May 85) pp 5-7

[Article by Academician N. F. Bondarenko, R. A. Poluektov, and V. P. Yakushev, VASKhNIL [All-Union Academy of Agricultural Sciences imeni V. I. Lenin (Order of Lenin, Order of Labor Red Banner)]]

[Text] Structure is suggested for an automated system for making plan and current decisions in programming harvests based on modern mathematical models of the production process of agricultural crops. Principles of system design and specifics of use of mathematical models at the various levels of agricultural production management are discussed.

In recent years, a number of dynamic simulation models of the production process of major agricultural crops has been developed [1, 3-5, 7, 8]. Implemented on computers, the models provide the capability of performing a series of alternative computations reflecting the process of producing a harvest under various specified weather conditions and agricultural technology, and assessing the response of plants to the conditions of the external environment. Success in modeling the production process allows setting the task of moving to the stage of using models directly in the practice of making plan and current decisions in programming harvests.

Let us note that until now, programming of harvests was based on accounting of so-called generalized soil and climatic indicators. Used in the process were parameters such as the bio-climatic potential (BKP), hydrothermal factor (GTK) et al., and the simplest regression models which related these values to the harvest.

The transition to using considerably more complex, but also more accurate, dynamic models requires fundamental revision of existing systems. The minimal configuration of a system for making decisions, automated on the basis of computers, must include at least three components: a software system which implements the models and algorithms for generating decisions on taking particular agricultural technological measures; data base organization and management in the form of an appropriate data base and data base

management system; and software affording an end user communication with the system (preferably in the interactive mode).

That is the way, in particular, the Pogoda [weather] automated system of agricultural meteorological and technological recommendations has been organized [2, 6, 9]. The system enables output of various reports on current weather conditions, on conditions and course of vegetation in the fields for the harvest being programed in the current year, the solution to a number of forecasting problems (planting schedules, onset of foehn phases, etc.), and the computation of values of the climatically conditioned harvest, quality index of soils, the harvest being programmed in each field (contour), and rates of mineral and organic fertilizers for harvest being programmed. The information includes four blocks: regional data, information on individual farms, a file on fields and special information. Finally, the system has a query language which allows generating requests for solution to problems both in the prescribed mode of system operation (for example, in drafting plan quotas), and in the mode of obtaining reports and forecasts directly during the growing season.

Let us examine how this system should be modified when simulation models are included in it. A flowchart of the modified system is shown in the figure. The right part reflects the functions listed above. The dynamic models have been included in the left part. Using simulation dynamic models of the production process substantially expands system capabilities, results in change of existing algorithms, and what is more important, affords the capability of solving fundamentally new problems. This is due primarily to features of simulation models. In particular, considered in them are the balances on the basic components of energy and mass exchange in the soil-plant-atmosphere system which allows quantitatively describing the processes of accumulation, transformation and loss of basic organic and mineral components from the system; forming of the harvest in the time from planting and right up to harvesting is described; this allows assessing the effect on this process of agricultural technological measures, schedules for implementing them, and the meteorological situation in the current season.

All this allows reproducing within the scope of the simulation system the sequence of change of states of the object being modeled, an agricultural field, under conditions as close to actual as possible.

Let us look at the specifics of making plan and current decisions in programming harvests based on dynamic models. There are two ways. The first deals with organization of experiments in which an exhaustive search of strategies of management of an agricultural ecosystem against the background of varying weather conditions is performed on a computer. However, considering the relatively large amount of machine time for these computations (a run for one growing season on the YeS-1035 computer, for example, takes 10 to 25 minutes), it is hardly possible at the present time to perform them for all the fields on a farm, and especially in the on-line mode. This method can be recommended for solving fundamental problems concerning strategies of agricultural development. In particular, simulation models should primarily be used

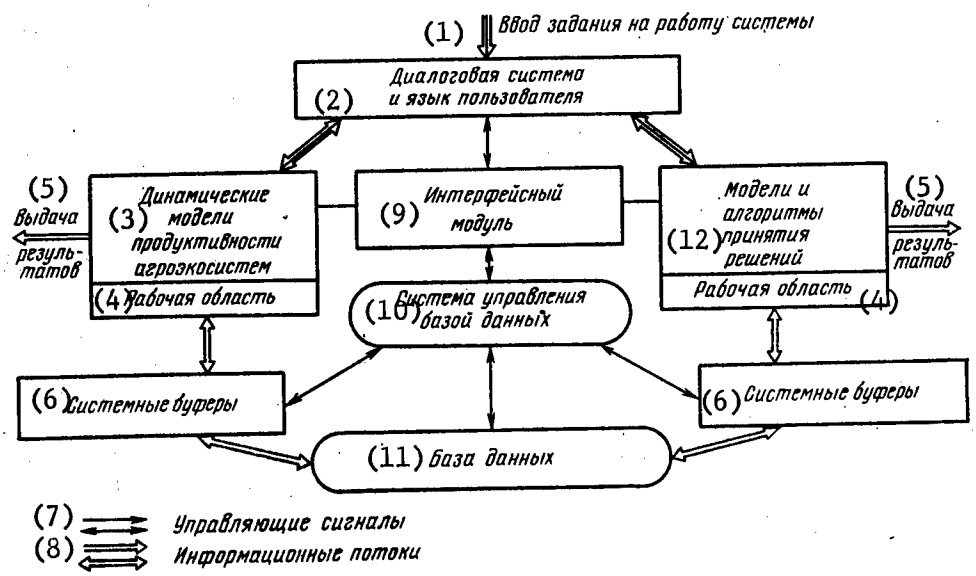


Fig. Structure of automated system for making decisions

Key:

1. job entry for system operation
2. interactive system and user language
3. dynamic models of productivity of agricultural ecosystem
4. work area
5. output of results
6. system buffers
7. control signals
8. information flows
9. interface module
10. data base management system
11. data base
12. models and algorithms for making decisions

to develop schemes for technologies conserving resources and energy within the scope of regional agricultural systems.

In connection with this, another, more efficient approach has to be developed. Within its scope, the basic dynamic model should be considered a knowledge base. Along with the information concentrated in the data base, it creates the basis for generating considerably simpler, specialized models including the simplest regression, similar to those which are already now widely used in reclamation, agricultural chemistry, and agricultural meteorology. These simple models make up the external part of the software for the automated system (the right part in the flowchart), while the simulation model can be considered its internal content (left part of flowchart). It is precisely on the basis of these external models that all problems of interest to the end user should be solved. In the process, the internal base model is used to determine the fields of application of external models and assessments of their constituent empirical factors.

We know, for example, that the balance method of computation of rates of fertilizers is the most promising. At the same time, practical use of it is running into almost insurmountable difficulties due to the lack of data on the factors of use of nutrients from the soil and fertilizers by plants. The reason is that factor values vary within a wide range when both weather conditions and agricultural technology change. The simulation model can be used for detailed study of these relations which will allow building a firm, quantitative base for the balance method of computation of fertilizers.

Improving the method of simulation modeling of productivity of agricultural ecosystems requires reexamination of the structure of automated information reference and information advisory systems used in programming harvests, and their algorithms and software. In the process, dynamic simulation models should be used in two ways. They should be directly included in a system for making decisions when considering too large a number of alternatives is not required, for example, at the level of substantiating regional recommendations in agricultural systems. When, however, tens of thousands of alternatives have to be considered (in particular, for individual fields and contours of farms), specialized, simplified models for which the simulation model can be considered the knowledge base must be used. Considering the variety of problems to be solved in agriculture and the hierarchical structure of management in agriculture, a hierarchical structure of automation of management should also be implemented by using distributed hardware (Unified System, mini and micro computers) compatible in information exchange, algorithms and software.

BIBLIOGRAPHY

1. Bondarenko, N. F. et al., "Modelirovaniye produktivnosti agroekosistem" [Modeling Productivity of Agricultural Ecosystems], Leningrad, Gidrometeoizdat, 1982.

2. Bondarenko, N. F. and Zhukovskiy, Ye. Ye., "Automated System for Agricultural Meteorological Information and Recommendations (Weather Project)," NAUCHNO-TEKH. BYUL. PO AGRONOMICHESKOY FIZIKE, No 32, 1977.

3. Galyamin, Ye. P. and Siptits, S. O., "Dynamic Model of Production Process and Use of It to Optimize Water Conditions," TRUDY INSTITUTA EKSPERIMENTALNOY METEOROLOGII, No 8 (67), 1977.

4. Gonchar-Zaykin, P. P., "Classification of Methods of Programming of Harvests and Fertility of Soils," NAUCHNO-TEKH. BYUL. PO AGRONOMICHESKOY FIZIKE, No 57, 1984.

5. Poluektov, R. A. et al., "Imitatsionnaya model razvitiya agrofitotsenoza (preprint)" [Simulation Model for Development of Agricultural Phytocenosis (Preprint)], Moscow, VNIISI [All-Union Scientific Research Institute of Sanitary Testing], 1984.

6. Poluektov, R. A. and Zhukovskiy, Ye. Ye., "Simulation Modeling of Agricultural Ecosystems and Its Information Base," in "Problemy ekologicheskogo monitoringa i modelirovaniye sistem" [Problems of Ecological Monitoring and Modeling of Systems], Vol 3, Leningrad, Gidrometeoizdat, 1980.

7. Polevoy, A. N., "Teoriya i raschet produktivnosti selskokhozyaystvennykh kultur" [Theory and Computation of Productivity of Agricultural Crops], Leningrad, Gidrometeoizdat, 1983.

8. Sirotenko, O. D., "Matematicheskoye modelirovaniye vodnoteplovogo rezhima i produktivnosti agroekosistem" [Mathematical Modeling of Water and Heat Conditions and Productivity of Agricultural Ecosystems], Leningrad, Gidrometeoizdat, 1981.

9. Yakushev, V. P., "Razrabotka i sozdaniye avtomatizirovannoy informatsionnoy sistemy dlya agrometeorologicheskogo obespecheniya programmirovaniya urozhayev" [Development and Establishment of an Automated Information System for an Agricultural Meteorological Base in Programming Harvests], Leningrad, 1982.

COPYRIGHT: VO "Agropromizdat", "Doklady VASKHNIL", 1986

8545
CSO: 1863/15

UDC 621.391.15:519.2

NOISELESS CODING OF COMBINATORIAL SOURCES, HAUSDORF DIMENSIONALITY AND KOLMOGOROV COMPLEXITY

Moscow PROBLEMY PEREDACHI INFORMATSII in Russian Vol 22, No 3, 1986
(manuscript received 15 May 84) pp 16-24

[Article by B. Ya. Ryabko]

[Abstract] A study is made of the problem of coding combinatorial sources, defined as arbitrary subsets of the set of all words infinite in one direction in a certain finite alphabet. Three statements are analyzed: No algorithmic limitations are placed on the code, coding and decoding are implemented by Turing machines, and coding and decoding are implemented by finite automata. The major result of the article is that the attainable lower boundary of cost of arbitrary codes is the Hausdorf dimensionality, an analog of the theorem of Shannon of noiseless coding of probability sources, with the Hausdorf dimensionality acting as the Shannon entropy. The main result indicates that the Hausdorf dimensionality is the lower boundary of cost of a code even when Shannon entropy does not exist. For the latter two statements of the problem, the lower cost boundaries are expressed through the Kolmogorov complexity and quasientropy. Optimal codes are constructed for sources generated by formal grammars. References 16: 9 Russian, 7 Western.

6508/5915
CSO: 1863/129

UDC 621.391.15

TWO CLASSES OF ALGORITHMS FOR DECODING BASED ON MINIMUM GENERALIZED DISTANCE

Moscow PROBLEMY PEREDACHI INFORMATSII in Russian Vol 22, No 3, 1986
(manuscript received 26 Apr 83; after revision 5 Oct 85) pp 35-42

[Article by S. I. Kovalev]

[Abstract] Decoding based on the minimum generalized distance includes a stage of repeated attempts to decode a sequence received with error correction and various combinations of erasures. The number of attempts may be as great

as [(d+1)/2]. Primary attention is given in this work to solution of the following problem: For a fixed number $l<[(d+1)/2]$ of decoding attempts, determine the combination of erasures for each attempt which minimizes the possibility of loss of the attainable correcting properties in comparison to the decoding procedure of Forney. Two classes of algorithms are studied: The class in which the set of l combinations of erasures for the ordered sequence is fixed and does not change from sequence to sequence; and the class in which the set of combinations of erasures for the ordered sequence is selected based on an analysis of the likelihood factors of the characters of the sequence. Figures 6, references 6: Russian.

6508/5915
CSO: 1863/129

UDC 621.391.1:519.28

SELF-ADAPTING ALGORITHM FOR MINIMAX NONPARAMETRIC ESTIMATION OF SPECTRAL DENSITY

Moscow PROBLEMY PEREDACHI INFORMATSII in Russian Vol 22, No 3, 1986 (manuscript received 29 Apr 84) pp 62-76

[Article by S. Yu. Yefroimovich and M. S. Pinsker]

[Abstract] A study is made of the problem of estimating the spectral density of a Gaussian steady sequence. An adaptive minimax nonparametric algorithm is utilized, assuming that the spectral density is a point on an ellipsoid in a Hilbert space with unknown axes. It is found that asymptotic minimax estimates of the spectral density can be generated for a broad class of sets, convenient for practical utilization. References 9: Russian.

6508/5915
CSO: 1863/129

UDC 681.333(536.21+536.631)

QUASIANALOGY AND THE PROBLEM OF HEAT CONDUCTIVITY

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86 (manuscript received 21 Nov 85) pp 83-87

[Article by Yu. M. Matsevityy]

[Abstract] Unsteady two- and three-dimensional heat conductivity problems are quite difficult to solve with ordinary computer systems due to their low speed, poor convergence of computation, complexity and insufficient accuracy of analog and hybrid processes. The method of quasianalogies can be used to model such nonlinear thermal processes. This article studies methods of solving heat conductivity problems using analog and hybrid analog-digital computer systems and the Kirkhoff and Goodman integral transform methods. Figures 3, references 21: Russian.

6508/5915
CSO: 1863/139

UDC 621.394.74

TRANSMISSION OF PACKETS BY BLOCKED NONMODIFIED RANDOM MULTIPLE ACCESS STACK ALGORITHM

Moscow PROBLEMY PEREDACHI INFORMATSII in Russian Vol 22, No 3, 1986 (manuscript received 12 Nov 84) pp 96-102

[Article by B. S. Tsyvakov and S. P. Fedortsov]

[Abstract] Previous works suggested a number of tree algorithms for random multiple access of packets in a shared channel with feedback. This article presents an upper bound for mean packet delay for one such algorithm, called (B, N, U, 2). The system studied consists of stations transmitting packets through a shared channel. The number of stations is large, assumed equal to infinity. All transmitted packets are one time unit in length. Where the input flow of packets has low intensity, the bound of maximum delay time is found to be linear. Figure 1, references 5: 3 Russian, 2 Western.

6508/5915
CSO: 1863/129

UDC 681.324

DEADLOCK AVOIDANCE IN TRANSMISSION OF INFORMATION IN COMPUTER SYSTEMS WITH PROGRAMMABLE STRUCTURE

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86 (manuscript received 11 Apr 84; after revision 30 Jul 84) pp 17-21

[Article by E. A. Manakhova]

[Abstract] Existing deadlock avoidance procedures place a priori limitations on possible packet transmission paths through a communication network, require complex structuring of buffer pools, and do not prevent livelocks. This article studies a decentralized algorithm for control of streams of packets in computer systems with programmable structure which prevents deadlocks but does not have these shortcomings. It guarantees no deadlocks upon transmission of messages regardless of the subsystem in which they were transmitted, and assure delivery of each packet in a finite time. It implements redirection of packets of the network to the file server and downward from the file server to a receiving computer with a maximum redirection path length of twice the

diameter of the transmitting subsystem. The maximum increase in transmission time caused by the algorithm is thus twice the time required to transmit a packet across the network. Figure 1, references 16: 9 Russian, 7 Western.

6508/5915
CSO: 1863/139

UDC 621.391.1:519.27

MEDIAN FILTRATION OF DETERMINISTIC AND STEADY RANDOM SIGNALS

Moscow PROBLEMY PEREDACHI INFORMATSII in Russian Vol 22, No 3, 1986
(manuscript received 27 Jun 84) pp 88-95

[Article by L. I. Piterbarg]

[Abstract] Results of a previous work by the same author, in which the concept of median filtration of a process with continuous time was introduced and the rate of convergence of statistical characteristics of an output signal to the corresponding characteristics of the input signal as the width of a window narrows to zero for certain Gaussian processes was found, are extended to arbitrary steady processes. The robustness of the median relative to a dwindling stream of pulsed noise is studied. References 3: Russian.

6508/5915
CSO: 1862/129

UDC 681.324

STRUCTURE AND INTERACTION OF PROCESSES IN DISTRIBUTED AUTOMATED DATA BASES

Kiev ELEKTRONNOYE MODELIROVANIYE in Russian No 5, Sep-Oct 86
(manuscript received 29 Apr 84; after revision 11 Feb 85) pp 91-93

[Article by I. N. Bacherikov]

[Abstract] The most efficient method of organizing information support in computer network is the distributed automated data base, an automated system for collection, storage, editing and retrieval of information. Each user accesses the data in such a network through the network and a data base management system. A structural diagram is presented of the processes which occur at a network node in providing user access to a distributed data base. The use of a high level query language allows processing of data either at the location where a query is made or at the location where the data are stored. Systems with distributed data base are found to have significant advantages both in set up and in operation over systems with concentrated data base. Figure 1, references 3: 2 Russian, 1 Western.

6508/5915
CSO: 1863/139

- END -