Abstracts of Current Computer Literature

CONTENTS

ABSTRACTS .................................................. Pages 672–680

Abstract Numbers

0) GENERAL; STANDARDS; EDUCATION................. 6814
1) LOGIC AND SWITCHING THEORY; SEQUENTIAL MACHINES ........................................ 6815–6828
2) DIGITAL COMPUTERS AND SYSTEMS .................. 6829–6833
3) LOGIC DEVICES AND CIRCUITS (HARDWARE) ...... 6834–6836
4) DIGITAL STORAGE AND INPUT-OUTPUT EQUIPMENT 6837–6843
5) PROGRAMMING AND CODING OF DIGITAL MACHINES 6844–6860
6) LINGUISTICS, DOCUMENTATION, AND HUMANITIES 6861
7) BEHAVIORAL SCIENCE, PATTERN RECOGNITION, AND ARTIFICIAL INTELLIGENCE .......... 6862–6868
8) MATHEMATICS ........................................ 6869–6878
9) PROBABILITY, MATHEMATICAL PROGRAMMING, DIGITAL SIMULATION, INFORMATION THEORY, AND COMMUNICATION SYSTEMS .................. 6879
10) SCIENCE, ENGINEERING, AND MEDICINE ......... 6880–6886
11) ANALOG AND HYBRID COMPUTERS ................... –
12) REAL-TIME SYSTEMS AND AUTOMATIC CONTROL; INDUSTRIAL APPLICATIONS ............ 6887
13) GOVERNMENT, MILITARY, AND TRANSPORTATION APPLICATIONS ......................... 6888
14) BUSINESS APPLICATIONS OF INFORMATION PROCESSING .................................. 6889–6890

DESCRIPTOR-IN-CONTEXT INDEX .......................... Pages 680–685
IDENTIFIER INDEX ........................................ Pages 686–687
AUTHOR INDEX ............................................. Page 688
0) General; Standards;
Education


This proposed USA Standard presents control procedures for data communication systems that employ the USA Standard Code for Information Interchange (USASCII). It is intended for systems that are controlled through the use of the communication control characters provided in USASCII. Other standards have been adopted or are being developed which prescribe the character structure, bit sequencing, signaling speed, message format, and other parameters vital to the communication of information in a data system. In the development of this standard both historical and present data communication practices were considered. In addition, consideration was given to the requirements for future systems and to the applicable standards being developed internationally.

1) LOGIC AND SWITCHING THEORY; SEQUENTIAL MACHINES

6815 Two Level Realization of Switching Functions with a Minimum Number of Threshold Gates, B. D. Carroll and C. L. Coates (Texas U., Austin); Rept. TR-56, 83 pp., September 1966; CFSTI, AD 078 106, $3.00.

The paper deals with the synthesis of networks of threshold gate logic elements which realize nonlinearly separable switching functions and which contain two levels of logic and a minimum number of logic elements. An algorithm is developed which can be used to obtain the desired network for a given switching function. The function may be incompletely specified. The algorithm is based on the realization procedures of Coates and Lewis. Necessary and sufficient conditions for removing an inconsistency are developed. A Boolean function technique for applying the removal rules to a set of inconsistencies is given. This technique consists of defining a Boolean function called a constraint function for each inconsistency that has been encountered. The product of constraint functions indicates the possibilities for defining gates to remove the inconsistencies. A zero product of constraint functions indicates that the set of inconsistencies cannot be removed with the number of gates being used.

6816 Logical Design of an Optimal Network by Integer Linear Programming: Part I, S. Muroga and T. Ibaraki (Illinois U., Urbana); pp., July 1968; CFSTI, PB 180 029, $3.00.

Logical design of an optimal network by integer linear programming is formulated. Various restrictions on a network are easily incorporated and how to incorporate them is discussed. The number of gates, the number of levels, or one of other parameters of a network can be minimized. A network of multiple outputs is also easily treated. The formulation is based on threshold gates, since threshold gates are the generalization of conventional switching gates, such as AND, OR, and NAND gates. The formulation of designing networks composed of threshold gates can be easily derived from this general formulation. As an example, design of an optimal network with NOR gates is discussed in some detail. Computer programs to design optimal networks with NOR gates for all Boolean functions of three variables are available.


In the ever-increasing trend to relieve man of time consuming menial tasks via automation, the digital logic designer’s tasks have come under study. Various phases of this work are now being performed by computers. One phase of simulation of digital designs, is the topic of this paper. A computer program now exists which accepts as input a description of a digital logic schematic and from this input generates timing diagrams, loading statistics, propagation times, cross references, and various lists of the input data for future documentation. A sample of the elements which can be simulated are NAND gates, JK flip-flops, AC flip-flops, delay lines, one shots, and an 18-bit, 4K computer memory. The first part of the paper is oriented toward the logic designer who has no knowledge of the workings of a computer. For the computer programmers or for anyone interested in how the simulation is implemented on the computer, there is a short section at the end which shows some of the details of the actual computer program.


Factoring techniques are incorporated in computer-oriented algorithms for the synthesis of fan-in limited NAND switching networks. Tree networks with reduced gate count or levels of logic are sought. While example FORTRAN programs emphasize computer execution of the algorithms, they are also efficient for hand execution.


An important aspect of the packaging of digital networks is the allocation of logic gates to modules such that a predetermined objective function is minimized. In order to develop techniques for partitioning of a logic network the following problem is considered. Given an acyclic combinational network composed of various primitive blocks such as NOR gates, assume that a maximum of M gates can be “clustered” together into larger modules, and that a number of P modules are replicated in each larger module. Assume also that in a network composed of such larger modules, no delay is encountered on the interconnections linking two gates internal to a module, and a delay of one time unit is encountered on interconnections linking two gates in different modules. Find an easily applied algorithm that will result in a network such that the maximum delay through the network is minimized. Efficient, easily applied algorithms for optimal clustering are presented for the case in which the combinational network is in the form of a tree. For the more general case in which the combinational network is represented by a directed acyclic graph which is not a tree, an algorithm for optimal clustering is obtained, but subject to the provisions that the primitive blocks of the network may be replicated.

6820 Efficient Numerical Analysis of Nonlinear Circuits, D. A. Calahan (Michigan U., Ann Arbor); Rept. AFOSR-68-2142, 12 pp., October 1968; CFSTI, AD 677 768, $3.00.

The advantages of using sparse matrix methods and high order implicit integration algorithms for the transient solution of nonlinear switching circuits is shown. Computing speed improvements of factors of 100 are displayed for a typical switching circuit.


An algorithm which reduces the number of gates and connections (diodes) in two-level, multiple-output combinational logic networks is presented and compared with conventional minimization procedures.


It is proved that there exist context-free languages L1, L2, L3, and words w1, w2 such that it is recursively unsolvable to determine for an arbitrary word w(1~4) whether w1L1\w2L2\w3L3\w4L4, is the empty set (a finite set, a regular set, a context-free language, respectively), (5~8) whether w1L1 is the empty set (a finite set, a regular set, a context-free language, respectively), (9) whether w1L1 is a regular set, (10) whether w1L1\w2L2\w3L3\w4L4, and (11) an on-line recognizer of a set \lambda of words. Suppose that \lambda is given an input word wu. Let \lambda(\nu) denote the configuration of M immediately before it reads the first letter of w. M is said to be nonredundant if \lambda(\nu) = \lambda(\nu') for all words w, w' such that w = w' X. From unsolvable problem (11) above it follows that there exists a context-free language that has no nonredundant on-line recognizer.

An algorithm is presented which will determine whether any string \( w \) in \( 2^* \), of length \( n \), is contained in a language \( L \subseteq 2^* \) defined by a two-way nondeterministic pushdown automation. This algorithm requires time \( n^2 \) when implemented on a random access computer. It requires \( n^2 \) time and \( n^2 \) tape when implemented on a multitape Turing machine. If the pushdown automation is deterministic, the algorithm requires \( n^2 \) time on a random access computer and \( n^2 \log n \) time on a multitape Turing machine.


A study is made of the classes of predicates accepted by three types of multitape Turing machine. In order of decreasing acceptance powers, these are the general Turing machine, the linear-bounded automaton, and the two-way multitape non-writting automaton. Each class is shown to consist of all and only those predicates which can be defined by a corresponding class of predicate calculus formulas based on catenation, and involving as logical operators conjunction, disjunction, and a type of transitive closure on predicates of \( 2n \) variables.


This paper investigates the concepts of relational homomorphisms and their closely associated concepts of generalized congruence relations on automata which are in general incomplete, nondeterministic, and infinite. The concept of generalized isomorphism, which is a natural extension of the isomorphism concept in dealing with nondeterministic automata, is also studied.

6826 Synchronization and Decomposition of Finite Automata, W. F. Cutilp and W. L. Kilmer (Michigan State U., East Lansing); Rept. Scientific-2, 97 pp., August 1968; CSFTI, AD 677 759, $3.00.

An input sequence \( x \) synchronizes a finite automaton \( A \) if the state of \( A \) after receiving \( x \) is independent of the state of \( A \) before receiving \( x \). The automaton \( A \) is synchronizable if such a sequence \( x \) exists. The presentations of synchronizability and properties of the set of synchronizing sequences, both for arbitrary and particular classes of automata, motivate much of the present work. The homing and distinguishing problems are briefly discussed, with references to some of the related published research. The set of tapes which synchronize a purely \( k \)-definite automaton is characterized. This characterization is shown to carry over, but with a quite different proof, to ultimate-definite automata; and it is shown that every definite automaton is synchronizable.


A general formulation of automata is given which is similar to that of sequential machines introduced by Santos and Wee (1968). It is shown how the existing deterministic, nondeterministic, and probabilistic automata fit into this formulation and various other interesting ultimate-definite automata are given. The maximin automata, which is a particular class of the general definition, includes both deterministic and nondeterministic automata as special cases. Moreover, its state transition function as well as initial distribution may be interpreted as grade of membership functions of fuzzy sets. Despite its generality, most of the basic concepts of existing automata, e.g., equivalences, reduction, behaviors, etc. may be carried over to maximin automata with appropriate modifications. Moreover, results related to these basic concepts are generalizable to maximin automata. The resemblance between maximin automata and probabilistic automata (Paz, 1966) is also considered.

Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition — see 6865.

6828 An Algorithm Method for Organizing Checking Experiments, G. Gonenc (Polytechnic Inst., Brooklyn); Rept. PIBEE 68-0006, 63 pp., June 1968; CSFTI, AD 678 158, $3.00.

In this work an algorithm for organization of checking experiments for synchronous sequential machines possessing distinguishing sequences is given. The algorithm is based on the transition checking approach and yields an upper bound on the length of the checking sequence smaller than those given in previous works. The actual length of the experiments is, in general, well under the bound. A method to further decrease the experiment length is described.

2) DIGITAL COMPUTERS AND SYSTEMS


With a suitable adder organization it is possible to overlap the adder operation during a binary multiplication and significantly decrease the overall multiplication time. The method is explained and a prototype multiplier described. The new technique provides a very economical method of obtaining a reasonably fast multiplier.


A fast method of generating bit-reversed addresses for the fast Fourier transform is described.


This paper focuses attention upon the design of a processor and memory system which is structured to achieve a satisfactory balance of processor speed and memory speed when both the processor and input-output controller are simultaneously competing for memory service. A mathematical model is developed to investigate the degree to which the processor is capable of overlapping memory references with instruction execution as a function of respective cycle times, the number of independent memory modules, and input-output traffic. Utilizing this model, design tradeoffs and performance indices are graphically examined for a hypothetical system.

Logic Design Automation — see 6821.

The literature on computer reliability is replete with very convincing arguments for the need and the use of self-repair techniques, as a viable approach to significantly enhancing the reliability of both maintainable and nonmaintainable computers. However, it would seem that no comprehensive and coherent program for the development and optimal employment of such techniques exists. This means that no method exists in the open literature for deciding the following: 1) what self-repair techniques, taken singularly or in combination, provide the greatest improvement in reliability; 2) what methods are optimum for initiating fault diagnosis and self-repair by redundancy and replacement; 3) what constitutes a closed set of self-repair techniques and what theory can be formulated to demonstrate the set's completeness; and 4) what is the effect of self-repair on the total system relative to design, maintenance, availability, and so forth. Although several investigators have addressed themselves to some of the subproblems presupposed by items 1) through 4), no one seems to have considered the possibility of resolving such problems by the development and methodical employment of a comprehensive systems-effectiveness measure. Such an approach, together with some of its results, is described in this paper.

Checking Sequences for Synchronous Sequential Machines—see 6828.

3) LOGIC DEVICES AND CIRCUITS (HARDWARE)

6834
An Integrated Circuit Technique for Saturati

A limited-saturation device technique found to improve the absolute value, spread, and temperature dependence of storage times in integrated transistors, and compatible to existing integrated-circuit processing is described.

6835

A new equivalent circuit for the insulated-gate field-effect transistor (IGFET) is described. This device model is particularly useful for computer-aided analysis of monolithic integrated IGFET switching circuits. The results of computer simulations using the new equivalent circuit are in close agreement with experimental observations. As an example of a practical application, simulation results are shown for an integrated circuit IGFET memory cell.

6836

The experimental results are reported for a series of measurements made on a single-crystal gadolinium garnet, with a view to obtaining the characteristics which appeared necessary for the use of garnets as memory elements. In particular, variations in the hysteresis loop as a function of the temperature are discussed in order to obtain the dependence of the rectangularity on the temperature. The rectangularity has been found approximately equal to 1 from 5 to 20°C. The variations in switching time as a function of the amplitude and the rise time of the magnetic excitation field and as a function of the temperature have also been studied. Switching time of 1 μs can easily be obtained. Evidence is also shown of a dependence of the coercive field upon the rise time of the driving signal.

4) DIGITAL STORAGE AND INPUT-OUTPUT EQUIPMENT

Computer-Aided Design of an 18-bit, 14K Computer System—see 6817.

Analysis of Processor "Look-Ahead" and Simultaneous Operation of a Multi-Module Main Store—see 6832.

Organizing Matrices and Matrix Operations for Paged Memory Systems—see 6850.

Set-Theoretic Data Structure Suitable for Paging in a Mass Memory System—see 6853.

Auxiliary-Storage Associative Graphics Data Structure for PL/1—see 6854.

Simulation of an Integrated Circuit IGFET Storage Cell—see 6835.

Switching Characteristics of Garnets Required for Their Use as Storage Elements—see 6836.

6837

The requirements, design, and performance of an integrable dc coupled digit detector array for high-speed memory systems such as the plated wire is described. The array consists of 12 detectors on 100-mil centers, on a tantalum film glass substrate using beam-lead monolithic chips. There are 218 resistors and 144 beam lead transistors formed or bonded with an area of 0.8×1.2 inch. Ninety-six of the transistors are monolithic differential pairs. The detectors dynamically sense minimum bipolar 1 and 0 signals of ±1 mV within a longitudinal noise of 150 mV, and generate a two-rail logic output of 3 volts in 15 ns. This detection can take place within 20 ns of a 5-volt differential or common mode waveform.

6838

A 4-K-word 50-bit-per-word plated wire main memory was constructed and tested at a 200-ns cycle time. 32 matrix planes were constructed of the woven wire type (each plane contained 128 words and was stacked into 2 blocks for a total capacity of 4-K words). Special design features for high-speed operations are 1) memory stack construction with memory planes which are serially connected with grounded conductors, 2) balanced word line driving using transmission-line-type transformers, 3) bridge configuration of the digit-invert system, and 4) digit driving scheme using non-return-to-zero current. Operating margins of the system were tested with account being taken of the effect of magnetic domain wall creeping for more than 10^6 disturb pulses.

6839

A 12-MHz 6500-bit plated-wire memory system is described that shows feasibility for application in a time-compression multiplex analog transmission system. The requirements for an assumed ten-channel time-compression multiplex system (TCM) are that 720 words of 9 bits each be written sequentially at a 12-MHz rate and the 720 words be read out at the same rate but in a different sequence. This process of serial write and serial read is continuous as coded samples are taken from each of the ten input channels. The important properties of the memory system described in detail are the generation and steering of the one-ampere word current of 30-ns duration, the effect of the short-duration word current on memory write, and the signal-to-noise ratio achieved using beam-lead diode matrix word and beam-lead tantalum film digit detectors.

6840

Methods are described for reducing shot noise in magnetic-film memories which are to be read magnetooptically with the auxiliary aid of an electron beam. In such memories the principal source of shot noise is from the light which is used to illuminate the memory array. The following methods of reducing array shot noise are considered: 1) selective background, 2) magnetooptical balance, 3) temperature control of magnetooptical spectra in rare-earth iron garnets (REIG). It is concluded that the use of REIG spectra appears to offer not only the best but also a satisfactory solution to the array shot-noise problem.

6841

A random-access magnetic-film memory with a desired bit density of 10^6/cm^2 is postulated with writing and reading each to be performed with a deflectable electron
This simulation is intended for use at low, intermediate, and high bit densities. While simpler algorithms can be found for use with either very low bit densities, or with very high bit densities, they lack the generality needed for this purpose. The simulation is composed of three distinct sequential computations. First, using a noninteracting particle idealized M-H model and the arctangent head field formula, the tape magnetization is computed at 40 points per bit length in each of 5 laminae. Second, this magnetization is averaged for the mirror and impact scanning. Finally, each harmonic component is weighted according to a demagnetizing-remagnetizing factor given previously, and the final output voltage waveform is computed. Linearity and superposition are thus assumed for all processes following the obviously nonlinear record mechanism. Computed outputs are compared with experimental results for both single transition and multiple transition inputs. The computed and measured isolated output pulses differ by no more than 10 percent, without the adoption of physically unreasonable parameters. Output signals were computed for multiple transition inputs up to 20 000 flux reversals per inch (fr/in). These compare well with experimental results up to 15 000 fr/in.


The present status of interactive graphic displays in the application environment is reviewed. Although graphics data processing is still largely experimental, several applications have come into productive use especially in the areas of data and design analysis. The primary benefit from enhancing such applications with graphic displays is the savings in calendar time. Before surveying several application areas, user aspects and application characteristics are discussed.

Problem-Oriented Language for Handling Three-Dimensional Geometric Information—see 6855.

This paper discusses the technique of structural induction for proving theorems about programs. This technique is closely related to recursion induction but makes use of the inductive definition of the data structures handled by the programs. It treats programs with recursion but without assignments or jumps. Some syntactic extensions to Landin's functional programming language iewm are suggested which make it easier to program the manipulation of data structures and to develop proofs about such programs. Two sample proofs are given to demonstrate the technique, one for a tree sorting algorithm and one for a simple compiler for expressions.

Documentation for Digital Logic Design Programs—see 6817.

This paper describes the implementation on the London University Atlas computer of Bell Telephone Laboratories low level linked list language L6. A syntactical definition of L6 is given in terms of BCL, a general purpose programming language with special emphasis on data structures. The description of the implementation in BCL includes details of the general field handling routines.

Languages of String-Form Encoded Line Patterns—see 6865.

Compiler and Language for Chemical Plant Design and Simulation Programs—see 6883.

This report is the third of a set of three reports documenting work in the area of automatic compiler generation. The first two reports described the theoretical basis for such a system. This report documents an operating system embodying the concepts described in the first two volumes. The system described in this report allows a programmer to write in FORTRAN, ALGOL, or JOVIAL and produce object code for either the CDC 1604B or the UNIVAC 1218. The system described can be expanded to incorporate other machines or languages.

6847 LAP-6 Handbook, M. A. Wilkes (Washington U., St. Louis); Rept. TR-2, 49 pp., May 1967; CFSTI, PB 180 186, $3.00.
LAP-6 is an on-line program for the 1948-word LINC which used the LINC keyboard and scope for communication with the user, and the magnetic tapes for storage and working area. It may be used for preparation and editing of any character string or for LINC program preparation. LAP-6 handles the manuscript display in such a way that any portion of the manuscript can be displayed at any time and edited directly by simply adding or deleting lines. Changes are shown integrated with the manuscript display as the user types. Meta commands provide automatic filing of manuscripts and programs on LINC tapes and handles the conversion and memory loading of LINC binary programs. Debugging aids include displays of symbol tables and errors, and selectable access to the manuscript display for editing and converting. Meta commands may be added by the user to suit his needs.

An approach to system interfaces for high level languages using basic input/output
support facilities is described. It is shown that this technique can provide potentially inexpensive methods for programs to communicate with deeply embedded facilities such as command language processors.

6849
An Objective Function for Computer Systems Scheduling, H. Bielowski (Illinois U., Urbana); Rept. BCL-4.2, 23 pp., October 1968; CFSTI, AD 678 164, $3.00.

The obvious scheduling goal is formulated in terms of the maximum long-term revenue of an installation. The consequences for the scheduling operations are derived by introducing the concept of "revenue functions," which represent the contribution of each job and its dependence on the response time. The application of this criterion, together with some priori information on the computing tasks, could enable the system to adapt its behavior to various job mixes. The drawbacks of the common scheduling and priority systems are reviewed in the light of the proposed revenue analysis.

Mathematical Model of Overlapping Memory References with Instruction Execution—see 6832.

6850

Matrix representations and operations are examined for the purpose of minimizing the page fault occurring in a paged memory system. It is shown that carefully designed matrix algorithms can lead to enormous savings in the number of page faults occurring when only a small part of the total matrix can be in main memory at one time. Examination of addition, multiplication, and inversion algorithms shows that a partitioned matrix representation (i.e., one submatrix or portion per page) in most cases induced fewer page faults than a row-by-row representation. The number of page-pulls required by these matrix manipulation algorithms are also studied as a function of the number of pages of main memory available to the algorithm.

6851

The integrity of a mass storage file system can substantially exceed that of the hardware and software involved. This effect has been obtained at Cambridge University where such a file system forms an essential part of a multiaccess system. A fixed disc store is backed up with magnetic tape and this configuration forms the basis of a design study which seeks to identify the problems and principles involved in file management. The result is simple but effective and involves a two tier file dumping system.

6852
A General-Purpose Program for Manipulating Formatted Data Files, R. C. McLeeon and J. Hilsenrath (NBS, Washington, D. C.); 31 pp., August 1968; GPO, Washington, D. C., $0.40.

A program listing and description is given of REFORM, an independent program which is to manipulate and edit files containing as many as nine different fixed-field card formats. It can select or abridge information from any of the cards and print that information, or reformat new cards in any desired order or arrangement. Provision is made for introducing as many as twenty-six arbitrary strings of characters, each of which may be up to seventy-nine characters in length, thereby permitting the insertion of labels, headings, or comments into the file. The program, which operates on the 1108 computer at NBS, is written in ASA FORTRAN, and care has been taken to reduce to a minimum the program changes required to make the program run on other computers.

6853
Description of a Set-Theoretic Data Structure, D. L. Childs (Michigan U., Ann Arbor); Rept. TR-39, 39 pp., August 1968; CFSTI, AD 678 454, $3.00.

This paper is motivated by an assumption that many problems dealing with arbitrarily related data can be expedited on a digital computer by a storage structure which allows rapid execution of operations within and between sets of datum names. Such a structure should allow any set-theoretic operation without restricting the type of sets involved, thus allowing operations on sets of sets of . . . ; sets of ordered pairs, ordered triples, ordered . . . ; sets of variable-length n-tuples, n-tuples of arbitrary sets, etc., with the assurance that these operations will be executed rapidly. The purpose of a set-theoretic data structure (STDs) is to provide a storage representation for arbitrarily related data allowing quick access, minimal storage and extreme flexibility. This paper describes an STDs with the above properties utilizing a general implementation suitable for paging in a mass memory system.

6854

A recent approach to representing relations between entities in a graphics data structure has been to store information as triples in the form Attribute (Object) = Value. This paper describes an associative technique for holding a universe of triples on auxiliary storage and then accessing a triple in response to an inquiry. The paper also shows how related operations have been performed—on an experimental basis—with PL/I as the language for the controlling program, using machine-language subroutines to perform only the basic functions on associative storage.

6855

This paper argues that there is a need for a problem-oriented language to handle three-dimensional geometric information, and proposes a set of language facilities that illustrate how this need should be met. The emphasis is on the facilities needed for describing solid objects and their placement in space, and for defining and operating on configurations of objects.

6856

A basic objective in the design of a package of general-purpose graphics subroutines was to make them accessible to FORTRAN programmers while circumventing some of the limitations of that language for graphics applications. This paper discusses how observation of graphics led to the establishment of design criteria for a subroutine package to facilitate the generation of interactive displays. It outlines how application programmers would use the package including procedures for communication between the console and the program. Many of the fundamental concepts that characterize the package are described, including provisions for display modification and animation.

6857

A graphic job processor that enables non-programmers to introduce application programs conversationally from a display console is described. Although not restricted to graphics applications, the processor makes the same display console available for both job definition and interaction with a graphics application program. This paper discusses some of the factors considered in designing displays to elicit information from the user. The structure of the processor is also described, including its interface to the operating system under which it functions, communication among the system operator, the console user, and the application programmer is also treated.

6858

This paper discusses a system called DISPLAY that interpretively executes FORTRAN statements entered at a display console, allowing graphics users to perform unanticipated computations and to more easily debug graphics application programs. The relationships among the operating system, the display terminal, and the computing system are discussed, and the major components of this system are described. A command language, the FORTRAN IV subset, and the graphics language provided for users are presented. Internal operation of the graphic facility is outlined.

6859

Discussed are two computer programs
ABSTRACTS OF CURRENT COMPUTER LITERATURE

for generating and realistically plotting any view of a three-dimensional object from the same object description, thereby simulating the viewpoints of a person moving around the object. Although the programs have been implemented—on an experimental basis—for digital plotting, the use of the underlying concepts for graphic display is contemplated. Involved in the Subject program are approaches to some of the most difficult problems in three-dimensional graphics—the hidden-line problem, approximating curved solids by polyhedra, and simulating degrees of surface transparency. The description of the program LEPER emphasizes the design of data storage for the object description. This scheme allows the use of the same data for generating all views of the object. The data structure can be modified to adjust the dimensions of the scene and the relative orientations of the component parts.


Modern digital computers in conjunction with microfilm plotters, are capable of generating a wide variety of images. Beyond the mere plotting of curves, graphs, and labels, special programming techniques can produce halftone renditions of pictorial material with wide applications in optical signal processing and data portrayal. Other possibilities include the composition of images from alphanumeric characters, the generation of moiré patterns, and the encryption of images with subsequent recovery by optical means using computer-generated "keys." More sophisticated techniques allow the combining in a single optical display of different kinds of information in hitherto unforeseen ways.

6861 Description of FORMAT, a Text-Processing Program, G. M. Berns (IBM, Wheaton, Maryland); Commun. ACM, vol. 12, pp. 141–146, March 1969.

FORMAT is a production program which facilitates the editing and printing of "finished" documents directly on the printer of a relatively small (64k) computer system. It features good performance, totally free-form input, very flexible formatting capabilities including up to eight columns per page, automatic capitalization, aids for indexing construction, and a minimum of context items. It is written entirely in FORTRAN IV.

Manuscript Display and Editing—see 6847.

Neutron Cross-Section Data Evaluation, Storage, and Retrieval—see 6881.

Geometric Relationships for Retrieval of Geographic Information Related to City Maps—see 6888.

7) BEHAVIORAL SCIENCE, PATTERN RECOGNITION, AND ARTIFICIAL INTELLIGENCE


Statistical recognition procedures can be derived from the functional form of underlying probability distributions. Successive approximation to the probability function leads to a class of recognition procedures. In this note a hierarchical method of designing recognition functions which satisfy both the least-square error property and a minimum decision error rate property is given. The discussion is restricted to a binary measurement space and its dichotomous classification.

6863 Feature Extraction in Pattern Recognition, J. T. Tou (U. Florida, Gainesville); Pattern Recognition, vol. 1, pp. 3–11, July 1968.

Sensing, feature extraction, optimum decision, parameter estimation, adaptation, and learning are the major problems in the design of pattern recognition systems. This paper discusses the selection of mathematical features on the basis of entropy minimization and introduces the concept of extracting statistical features by the method of kernel approximation.

6864 A Recursive Bayesian Approach to Pattern Recognition, H. M. Heisein (IBM, Rockville, Maryland); Pattern Recognition, vol. 1, pp. 13–31, July 1968.

The pattern classification problem is stated in terms of an ideal system and a model system. The ideal system gives the true classification of each input pattern while the model system gives an estimate of the probable classification of each pattern. The estimate is produced by the model as a parametrically defined function of the input pattern. The problem is to find a training algorithm which determines the values of the parameters, initially unspecified, from a sequence of inputs and outputs of the ideal and model systems, and, thereby, determines the characteristics of the model. A general theoretical solution to this problem is given by a recursive Bayesian estimation procedure. Individual implementations may be derived by approximations which trade accuracy for computational ease. Assumptions of linearity and normality lead to a perceptron-like algorithm. Several nonlinear and nonnormal cases are also considered.


By treating patterns as statements in a two-dimensional language, it is possible to apply linguistic theory to pattern analysis and recognition. In this paper, line patterns are encoded into string form using the chain code developed by Freeman. A class of patterns can readily be modified for use to a set of strings that is examined using theory that exists for string languages and automata. Pattern languages formed on the basis of equations in two variables and various properties are related to the hierarchy of string language classes. The known relationships between classes of string languages and classes of automata can then be applied to determine bounds on the time and memory required to recognize the various patterns. Results can be extended to other forms of pattern encoding provided that a suitable translator can be constructed.

6866 Distance Functions on Digital Pictures, A. Rosenfield and J. L. Pfaltz (U. Maryland); Pattern Recognition, vol. 1, pp. 33–61, July 1968.

This paper describes algorithms for computing various functions on a digital picture which depend on the distance to a given subset of the picture. The algorithms involve local operations which are performed repeatedly, "in parallel," on every picture element and its immediate neighbors. Applications to the detection of "clusters" and "regularities" in a picture, and to the dissection of a region into "pieces," are also described.


This paper discusses a system designed for interactive problem solving by use of a graphic console. Existing application programs can readily be used with a graphic display device, and the graphics programming can usually be done in a higher level language. Based on intermediate results, the order of execution of application given modules can be controlled from the console. The system description emphasizes the structure and generation of display formats for displaying output, for accepting user-defined commands, and for accepting data that is made accessible to the application modules. Also described is a generalized data structure and a set of experimental routines designed to adapt the structure to particular needs.


This paper describes a program for solving a class of "word sum" or "cryptarithm" puzzles by a heuristic tree searching method. Formally the problem is to solve a set of simultaneous linear inequalities with the variables taking integer values.

Use of Graphics in a Heuristic Approach to Problem Solving—see 6882.
8) MATHEMATICS


Two different algorithms for computing the generalized factorial function defined by Cooper (1966) are written in a FORTRAN-like language and then, without use of the principle of recursion induction (McCarthy, 1962) are proved equivalent.

Polynomial Curve Fitting When Abscissas and Ordinates Are Both Subject to Error, M. O'Neill (Independent Computer Services, Belfast, N. Ireland), I. G. Sinclair (Short Brothers, Belfast, N. Ireland), and F. J. Smith (The Queen's University, Belfast, N. Ireland); Computer J., vol. 12, pp. 52–56, February 1969.

An iterative method is described for the least-squares curve fitting of a polynomial to a set of points in two dimensions when both the abscissas and ordinates are subject to error and when the weights of all the readings are known. The process converges, in general, to a polynomial giving the exact minimum of the “weighted” perpendicular distances onto the curve. It is shown that in practice Deming's method gives a solution close to this optimum polynomial.


A method of local smoothing of noisy data by making a least squares fit to a suitably chosen polynomial is described. Various \( n \)-dimensional formulas are derived and their effects compared empirically.


An algorithm for nonlinear minimax approximation is described, and shown to be convergent under conditions which are often assumed in practice. The algorithm is illustrated by the calculation of several approximations to the solution of the Blasius equation.


Some methods of successive approximation for the solution of simultaneous linear equations are discussed. The coefficient matrix \( A \) of the linear system is assumed to be sparse. It is shown that savings in the computer storage and the computing time are possible, if there exists a subset of the rows (columns) of \( A \), consisting of only orthogonal rows (columns). Such savings are also possible, if for some permutation matrices \( P \) and \( Q \), \( P A Q \) has a particular structure, i.e., singly bordered block diagonal form. It is shown that the set of orthogonal rows (columns) of \( A \), as well as \( P \) and \( Q \) can be determined by using some results from graph theory (e.g., incidence matrices, row and column graphs, points of attachment). Geometrical interpretations of the methods and their interrelationship are given.


The method of solving nonlinear simultaneous equations discussed consists of a particular form of Broyden's method used in conjunction with a particular form of Davidenko's method. The only information required to the user, apart from the procedure to compute the residuals, is an initial estimate of the solution.

Solution of a Set of Simultaneous Inequalities with Integer-Valued Variables—see 6868.

Use of High Order Implicit Integration Algorithms for Transient Solution of Nonlinear Switching Circuits—see 6820.


Most numerical methods for producing approximate Chebyshev series solutions to ordinary differential equations lead to a system of algebraic equations for the coefficients, while the truncation error (or a first order approximation to it for nonlinear equations) can be formulated as an infinite series. By solving the algebraic system with additional right-hand sides and by extrapolating the size of the exact coefficients from the computed ones, the first few terms in the error expansion result, giving an accurate error estimate varying with the independent variable unless the series is slowly convergent. The technique is illustrated by numerical examples.


Numerical integration of first-order conservative systems is also conservative. For practical computation this is a useful approximation within the effects of round-off error in application of the formulae. The result is invalidated when there is numerical instability and when finite arithmetic is used.


The stability of the Crank-Nicolson scheme for the numerical solution of the heat conduction equation subject to separated boundary conditions is demonstrated. This result is extended to separable equations with variable coefficients and to the heat conduction equation in a cylindrical geometry which has a singular coefficient. The solution of the difference approximation to the heat conduction equation is shown to reflect accurately the pattern of behavior of the differential equation, and this result is applied to the phenomenon of "persistent discretization error" in the solution to the difference equation.


A new digital computer method is developed for the Navier-Stokes equations. Finite differences, smoothing and a special boundary technique are fundamental. The method converges in practice for all Reynolds numbers. Examples illustrate both primary and secondary vortices and show the development of selected double-spiral equivorticity curves as the Reynolds number becomes infinite. As a special case, the method applies easily to biharmonic problems.

Graph Decomposition for Optical Logic Module Clustering—see 6819.

Graph Theory and Geometrical Techniques for Use in Contour Map Processing—see 6885.

9) PROBABILITY, MATHEMATICAL PROGRAMMING, DIGITAL SIMULATION, INFORMATION THEORY, AND COMMUNICATION SYSTEMS


An improved procedure is presented for generating orthogonal search vectors for use in Rosenbrock's and Swann's optimization methods. The new procedure shows considerable savings in time and in storage requirements, and deals more satisfactorily with certain cases in which the original method fails.

Recursive Bayesian Estimation Procedure for Pattern Recognition—see 6864.
ABSTRACTS OF CURRENT COMPUTER LITERATURE

Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming—see 6816.

Computer Simulation of Digital Design Logic—see 6817.

Computer Simulation of Unbiased Digital Recording—see 6842.

Simulation of Degrees of Surface Transparency—see 6859.

Compiler and Language for Chemical Plant Design and Simulation Programs—see 6883.

Modeling and Simulation of Insulated-Gate Field-Effect Transistor Switching Circuits—see 6835.

Modeling in Three Dimensions—see 6859.

Mathematical Model of Overlapping Memory References with Instruction Execution—see 6832.

Encryption of Images with Subsequent Recovery by Optical Means—see 6860.

Data Communication Control Procedures for USA Standard Code for Information Interchange—see 6814.

80-ns Plated-Wire Store for a Time-Compression Multiplex Transmission System—see 6839.

10) SCIENCE, ENGINEERING, AND MEDICINE

Computer-Aided Design of Digital Logic Circuits—see 6817.

Efficient Numerical Analysis of Nonlinear Switching Circuits—see 6820.

Computer-Aided Analysis of Monolithic Integrated IGFET Switching Circuits—see 6835.

Numerical Solution of the Heat Conduction Equation Subject to Separated Boundary Conditions—see 6877.


A group of programs, called SUMX, is used for statistically analyzing high-energy physics data by batch-processing techniques. Against this background, the paper discusses the first phase of a project directed toward placing the SUMX user on-line via a display console. On-line SUMX provides a helpful interactive mode of computer use in an inherently difficult application area. The experimental environment of the data source is discussed. Functions of component programs as well as the types of statistical analyses performed are presented.


Evaluation, storage, and retrieval of neutron cross-section data are of major concern to the International community of low-energy physicists. An experimentally evolving program, called SCORE, designed to perform these services is described, the overall concept of the SCORE program and the programming environment are presented; and cross section data entry, evaluation, and curve generation are discussed.


In demonstrating the advantageous use of graphics in a heuristic approach to problem solving, the deciphering of unknown crystal structures is used as an environment. The interactive graphic console lends concreteness to the crystallographer's abstract perception of three-dimensional structures. This paper describes how the crystallographer uses graphics in determining crystal structure by matching a known spatial molecular model with an experimental crystallographic model and by direct theoretical procedures when no model is available. Also presented are computer rendering techniques that make the man-machine relationship more productive. In conclusion, the author ponders the possibilities of more efficient and productive scientific research through the evolution of computer-aided data libraries.


A compiler, suitable for a language in which chemical plant design and simulation programs can be written, is described. It is suggested that eventually an extended ALGOL compiler will be most suitable for integrated plant design programs.


This paper describes a generalized program for the analysis and design of plate cam and follower mechanisms using a graphic display console. The experimental program was developed to study the use of interactive graphics systems for solving mechanical design problems. The program can handle almost all types of plate cam and follower mechanisms. A wide selection of motion curves permits the designer to specify any desired motion of a point of interest by synthesizing several curves into a displacement diagram.


Generalized techniques are developed whose use can simplify the solution of problems relating to contour maps. One of these techniques makes use of the topological properties of contour maps. The topology is represented by a graphical structure in which adjacent contour lines appear as connected nodes. Another generalized technique consists of utilizing geometrical properties to determine the characteristics of straight lines drawn on the contour map. Both of these techniques have been applied to the problem of locating the ground track of an aircraft from elevation readings obtained during a flight.


In this article some problems of formal diagnostics are formulated. Difficulties arising are discussed and a computer approach to medical and technical diagnostics is offered. The approach consists of organization of a process of automatic production of a decision rule on the basis of information stored in the set of parameters of correct classification of situations. A list of problems in medical and technical diagnostics is presented; these were solved by the author and his collaborators by an original method of learning, the so-called method of "Generalized Portrait."

12) REAL-TIME SYSTEMS AND AUTOMATIC CONTROL, INDUSTRIAL APPLICATIONS


Digital computer methods for PERT event time calculations may be based on the familiar methods used for hand computations, in which case the network must be topologically ordered. For the calculation of the earliest and latest event times it is sufficient to order the arcs in one of several ways. An ordering may be achieved by using the topological structure in a way analogous to the hand computation method. Alternatively, iterative methods may be used; these compare favorably provided the network data is presented in a suitable form.

13) GOVERNMENT, MILITARY, AND TRANSPORTATION APPLICATIONS


An experimental technique that has been developed for retrieving, by geometrical means, information related to city maps is reported. This paper emphasizes the analysis needed to translate a retrieval query into relationships among points, vectors, and polygons. An illustration of the technique is given.
of the project “Computers and the Small Firm,” being carried out by the University of Liverpool Data Processing Research Unit for the Ministry of Technology. Some preliminary results of the system design are presented.

This paper describes the second stage

**14) BUSINESS APPLICATIONS OF INFORMATION PROCESSING**

**6889**


**6890**


This paper discusses the need for formal notations and techniques for use in systems design and analysis. It argues that the data dictionary is a central tool in this work, makes proposals for formalizing it and suggests computer based procedures for aiding its construction.

**DESCRIPTOR-IN-CONTEXT INDEX**

**A**

Addition

- Binary Multiplication with Overlapped Addition Cycles 6830

Addresses

- Generation of Bit-Reversed Addresses for the Fast Fourier Transform 6831
  — see also Storage

Algebra

- see Algebraic Equations, Automata, Boolean, Matrices, Polynomials

Algebraic Equations

- Projecton Methods for Solving Linear Systems of Equations 6873
  - New Method of Solving Nonlinear Simultaneous Equations 6874
  - System of Algebraic Equations Resulting from Chebyshev Series Solution of Differential Equations 6875

Algorithms

- Use of High Order Implicit Integration Algorithms for Transient Solution of Nonlinear Switching Circuits 6820
  - Proof of a Tree Sorting Algorithm 8644
  - Matrix Manipulation Algorithms for Array Storage Allocation 6850
  - Recursive Bayesian Estimation Procedure for Pattern Recognition 6864
  - Perceptron-like Training Algorithm for Pattern Classification 6864
  - Equivalence of Two Algorithms for Cooper’s Generalized Factorial Function 6869

Algorithms for:

- Two-Level Realization of Switching Functions with a Minimum Number of Threshold Gates 6815
  - Logic Design Automation of Fan-In Limited NAND Networks 6818
  - Search for Tree Networks with Reduced Gate Count or Levels of Logic 6818
  - Allocation of Logic Gates to Modules for Minimum Delay 6819
  - Determining Time and Tape Complexity of Pushdown Automaton Languages 6823

- Organization of Checking Experiments for Synchronous Sequential Machines 6838
  - Computer Simulation of Unbiased Recording 6842
  - Minimization of Page Faulting in Paged Memory Systems 6850

Distance Functions on Digital Pictures 6866

Allocation

- Allocation of Logic Gates to Modules for Minimum Delay 6819
  - Matrix Manipulation Algorithms for Array Allocation 6850
  - Algebraic Equations — see Characters

Analysis

- Analysis of Processor “Look-Ahead” and Simultaneous Operation of a Multi-Module Main Store 6832
  - Analysis and Synthesis of a Self-Repairing Computer 6833
  - Computer-Aided Analysis of Monolithic Integrated IGFET Switching Circuits 6835
  - Data and Design Analysis Using Interactive Graphic Displays 6843
  - Revenue Analysis of Time-Sharing Computer Installations 6849
  - Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition 6865
  - On-Line Interactive Analysis and Display of High-Energy Physics Data 6880
  - Crystal Structure Analysis Using an Interactive Graphic Console 6882
  - Computer-Aided Data Dictionary for Systems Analysis Documentation 6890
  — see also Numerical Analysis

Approximation; Curve Fitting

- Feature Extraction in Pattern Recognition by Kernel Approximation 6863
  - Polynomial Curve Fitting When Abscissas and Ordinates are Both Subject to Error 6870
  - Local Smoothing of Noisy Data by Polynomials in u Dimensions 6871
  - Minimax Approximation in the Nonlinear Case 6872
  - Successive Approximation Method for Solving Simultaneous Linear Equations 6873
  — see also Error

Arithmetic

- Incrementing a Bit-Reversed Integer 6831
  — see also Addition, Multiplication, Subtraction

Arrays

- High-Speed dc Coupled Digit Detector

Array for Plated-Wire Storage Systems 6837

Improving the Signal-to-Noise Ratio of Photon and Electron Beam Accessed Magnetic Film Storage Array 6840

Matrix Manipulation Algorithms for Array Storage Allocation 6890

— see also Cells, Storage

Artificial Intelligence

- see Heuristic, Learning, Patterns, Problem Solving, Theorem Proving

Assembly

- On-Line List Assembly Program for 3048-Word LINC Computer 6847

Associative

- Auxiliary-Storage Associative Graphics Data Structure for PL/I 6854

Automata

- Determining Time and Tape Complexity of Pushdown Automaton Languages 6867
  - Classes of Automata and Transitive Closure 6824
  - Rational Homomorphisms of Automata 6825
  - Synchronization and Decomposition of Finite Automata 6826
  - Maximin Automata 6827
  - Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition 6865
  — see also Context-Free, Languages, Sequential Machines

Automated; Automatic; Automation

- Automated Design of Digital Logic Circuits 6818
  - Logic Design Automation of Fan-In Limited NAND Networks 6818
  - Search for Tree Networks with Reduced Gate Count or Levels of Logic 6818
  - Allocation of Logic Gates to Modules for Minimum Delay 6819
  - Determining Time and Tape Complexity of Pushdown Automaton Languages 6823
  - Organization of Checking Experiments for Synchronous Sequential Machines 6838
  - Computer Simulation of Unbiased Recording 6842
  - Minimization of Page Faulting in Paged Memory Systems 6850

Distance Functions on Digital Pictures 6866

Allocation

- Allocation of Logic Gates to Modules for Minimum Delay 6819
  - Matrix Manipulation Algorithms for Array Allocation 6850
  — see also Characters

Analysis

- Analysis of Processor “Look-Ahead” and Simultaneous Operation of a Multi-Module Main Store 6832
  - Analysis and Synthesis of a Self-Repairing Computer 6833
  - Computer-Aided Analysis of Monolithic Integrated IGFET Switching Circuits 6835
  - Data and Design Analysis Using Interactive Graphic Displays 6843
  - Revenue Analysis of Time-Sharing Computer Installations 6849
  - Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition 6865
  - On-Line Interactive Analysis and Display of High-Energy Physics Data 6880
  - Crystal Structure Analysis Using an Interactive Graphic Console 6882
  - Computer-Aided Data Dictionary for Systems Analysis Documentation 6890
  — see also Numerical Analysis

Approximation; Curve Fitting

- Feature Extraction in Pattern Recognition by Kernel Approximation 6863
  - Polynomial Curve Fitting When Abscissas and Ordinates are Both Subject to Error 6870
  - Local Smoothing of Noisy Data by Polynomials in u Dimensions 6871
  - Minimax Approximation in the Nonlinear Case 6872
  - Successive Approximation Method for Solving Simultaneous Linear Equations 6873
  — see also Error

Arithmetic

- Incrementing a Bit-Reversed Integer 6831
  — see also Addition, Multiplication, Subtraction

Arrays

- High-Speed dc Coupled Digit Detector

Array for Plated-Wire Storage Systems 6837

Improving the Signal-to-Noise Ratio of Photon and Electron Beam Accessed Magnetic Film Storage Array 6840

Matrix Manipulation Algorithms for Array Storage Allocation 6890

— see also Cells, Storage

Artificial Intelligence

- see Heuristic, Learning, Patterns, Problem Solving, Theorem Proving

Assembly

- On-Line List Assembly Program for 3048-Word LINC Computer 6847

Associative

- Auxiliary-Storage Associative Graphics Data Structure for PL/I 6854

Automata

- Determining Time and Tape Complexity of Pushdown Automaton Languages 6867
  - Classes of Automata and Transitive Closure 6824
  - Rational Homomorphisms of Automata 6825
  - Synchronization and Decomposition of Finite Automata 6826
  - Maximin Automata 6827
  - Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition 6865
  — see also Context-Free, Languages, Sequential Machines

Automated; Automatic; Automation

- Automated Design of Digital Logic Circuits 6818
  - Logic Design Automation of Fan-In Limited NAND Networks 6818
  - Search for Tree Networks with Reduced Gate Count or Levels of Logic 6818
  - Allocation of Logic Gates to Modules for Minimum Delay 6819
  - Determining Time and Tape Complexity of Pushdown Automaton Languages 6823
  - Organization of Checking Experiments for Synchronous Sequential Machines 6838
  - Computer Simulation of Unbiased Recording 6842
  - Minimization of Page Faulting in Paged Memory Systems 6850

ABSTRACTS OF CURRENT COMPUTER LITERATURE

Boolean
Designing Optimal Networks with XOR Gates for all Boolean Functions at Three Variables 6816

C
Calculus
—see Differential Equations, Integration
Cells
Simulation of an Integrated Circuit IGFET Storage Cell 6835
—see also Arrays
Characters
Character Structure Standards for Information Interchange 6814
Program for Preparing and Editing Character Strings 6847
Composition of Images from Alphanumeric Characters 6860
—see also Patterns, Recognition
Circuits
—see Gates, Integrated Circuits, Logic, Modules, Networks, Switching
Classification
Didotomous Classification of a Binary Measurement Space 6862
Automatic Production of Decision Rules for Diagnostic Classification 6886
—see also Patterns
Codes; Coding; Encoding
Data Communication Control Procedures for USA Standard Code for Information Interchange 6814
Operating System Which Produces Object Code for CDC 1600B or UNIVAC 1218 6846
Encryption of Images with Subsequent Recovery by Optical Means 6860
Languages of String-Form Encoded Line Patterns 6865
Combinational
Acyclic Combinational Networks Composed of Primitive Blocks 6819
Minimization of Gates and Connections in Combinational Logic Networks 6821
Communication
Data Communication Control Procedures for USA Standard Code for Information Interchange 6814
Methods for Programs to Communicate with Command Language Processors 6848
Compilers
Proof of a Compiler for Expressions 6844
Automatic Compiler Generation 6846
Compiler and Language for Chemical Plant Design and Simulation Programs 6883
—see also Programming Languages
Complexity
Determining Time and Tape Complexity of Pushdown Automaton Languages 6823
Computation
Efficient Numerical Analysis of Nonlinear Switching Circuits 6820
—see also Complexity
Computer Applications
Computer-Aided Design of Digital Logic Circuits 6817
Efficient Numerical Analysis of Nonlinear Switching Circuits 6820
Computer-Aided Analysis of Monolithic Integrated IGFET Switching Circuits 6835
Debugging of Graphics Applications Programs Using DISPLAYTRAN 6858
Graph Theory and Geometrical Techniques for Use in Contour Map Projection 6884
Methods of Computing Event Times in Project Networks 6887
Computer Use in Small Firms 6889
Computer-Aided Data Dictionary for Systems Analysis Documentation 6890
—see also Computer Systems, Control, Documentation, Mathematics, Patterns, Programs for, Retrieval, Simulation
Computer Center Management
Revenue Analysis of Time-Sharing Computer Installations 6849
Computer Systems
Optimally Reliable Computing Systems 6833
Objective Function for Computer Systems Scheduling 6849
Relationships Among the Operating System, the Display Terminal and the Computing System in DISPLAYTRAN 6858
—see also Computers, Digital Computers, On-Line, Operating Systems, Time-Sharing
Computers
Analysis and Synthesis of a Self-Repairing Computer 6833
On-Line List Assembly Program for 2048-Word LINC Computer 6847
—see also Automata, Computer Applications, Computer Center Management, Computer Systems, Digital Computers, Graphics, Processes, Sequential Machines, Time-Sharing
Consoles
Conventional Graphic Job Processor Using a Display Console 6857
System for Interpreting Executing FORTRAN Statements Entered at a Display Console 6858
System for Interactive Problem Solving Using a Graphic Display Console 6857
Crystal Structure Analysis Using an Interactive Graphic Console 6882
Cam Design on an Interactive Graphic Display Console 6884
—see also Terminals
Context-Free
Recursively Unsolvable Problems on Context-Free Languages with Application to On-Line Language Recognizers 6822
Control
Data Communication Control Procedures for USA Standard Code for Information Interchange 6814
Propagating Adder Control Signals in a Parallel Binary Multiplier 6830
Control of the Execution of Problem Solving Applications Modules on a Graphic Console 6867
D
Data
Data Communication Control Procedures for USA Standard Code for Information Interchange 6814
Data and Design Analysis Using Interactive Graphic Displays 6843
Programming Technique for Generating Half-Tone Images for Optical Signal Processing and Data Portrayal 6860
Local Smoothing of Noisy Data by Polynomials in n Dimensions 6871
On-Line Interactive Analysis and Display of High-Energy Physics Data 6880
Computer-Aided Data Dictionary for Systems Analysis Documentation 6890
—see also Data Management, Data Structures, Files, Information, Statistics, Trees
Data Management
Manipulation and Editing of Formatted Data Files 6852
Data Processing
—see Computer Systems, Digital Computers, Processors
Data Structures
Inductive Definition of Data Structures Handled by Programs 6844
General Purpose Programming Language with Special Emphasis on Data Structures 6845
Set-Theoretic Data Structure Suitable for Paging in a Mass Memory System 6853
Auxiliary-Storage Associative Graphics Data Structure for PL/I 6854
Data Structure for Graphic Display of Three-Dimensional Objects 6859
Generalized Data Structure and Programs for Interactive Problem Solving Using a Graphic Display Console 6867
Generalized Data Structure and Programs for Interactive Problem Solving Using a Graphic Display Console 6867
Data Transmission
—see Codes, Communication, Interchange Interfaces
Debugging
Debugging Aids for LAP-6 Program 6847
Debugging of Graphics Applications Programs Using DISPLAYTRAN 6858
Decision Processes
Recognition Functions which Satisfy Both Least Square Error and Minimum Decision Error Rate Properties 6852
Automatic Production of Decision Rules for Diagnostic Classification 6886
Decomposition
Graph Decomposition for Optimal Logic Module Clustering 6819
Synchronization and Decomposition of Finite Automata 6826
Delay
Module Clustering to Minimize Delay in Digital Network Interconnections 6819
Design
Data and Design Analysis Using Interactive Graphic Displays 6843
Hierarchical Method of Designing Statistical Recognition Functions 6862
Compiler and Language for Chemical Plant Design and Simulation Programs 6883
Cam Design on an Interactive Graphic Display Console 6884
—see also Logic Design, Realization, Synthesis
Diagnosis
Formal Methods of Diagnosis in Engineering and Medicine 6886
—see also Error, Failures
Differential Equations (Ordinary)
Error Estimation for the Solution of Ordinary Differential Equations in Chebyshev Series 6875
Numerical Integration of Conservative
Systems of First-Order Ordinary Differential Equations 6876

Differential Equations (Partial)
Numerical Solution of the Heat Conduction Equation Subject to Separated Boundary Conditions 6877
Digital Computer Method for Solving Navier-Stokes Partial Differential Equations 6878

Digital
Computer Simulation of Digital Design Logic 6817
Module Clustering to Minimize Delay in Digital Network Interconnections 6819
Distance Functions on Digital Pictures 6866
—see also Communication, Integrated Circuits, Plotting, Simulation, Switching, Systems

Digital Computers
Image Generation Using Digital Computers and Microfilm Plotters 6860
—see also Computer Systems, Computers, Processors

Digital Recoding
—see Magnetic Recording

Disc Store
File Management for a Fixed Disc Store Backed up by Magnetic Tape 6851

Displays
Implementation and Usage of Interactive Graphic Displays 6843
Manuscript Display and Editing 6847
DISPLAYTRAN Conversational Display System 6838
On-Line Interactive Analysis and Display of High-Energy Physics Data 6880
Cam Design on an Interactive Graphic Display Console 6884
—see also Characters, Graphics, Terminals

Documentation
Documentation for Digital Logic Design Programs 6817
Manuscript Display and Editing 6847
Text-Processing Program for Editing, Formatting and Printing Finished Documents 6861
Computer-Aided Data Dictionary for Systems Analysis Documentation 6890
—see also Information, Retrieval

Editing
Program for Preparing and Editing Character Strings 6874
Manipulation and Editing of Formatted Data Files 6852
Text-Processing Program for Editing, Formatting and Printing Finished Documents 6861

Electron Beams
Improving the Signal-to-Noise Ratio of Photon and Electron Beam Accessed Magnetic Film Storage Arrays 6840
Feasibility of Lorentz Readout of a High-Density Fast Magnetic-Film Store 6841

Equations
—see Algebraic Equations, Boolean, Differential Equations

Equivalence
Equivalence of Automata 6827
Equivalence of Two Algorithms for Cooper’s Generalized Factorial Function 6869

Error
Data Communication Error Control Procedures 6814

Recognition Functions Which Satisfy Both Least Square Error and Minimum Decision Error Rate Properties 6862
Polynomial Curve Fitting When Abscissas and Ordinates are Both Subject to Error 6870
Error Estimation for the Solution of Ordinary Differential Equations in Chebyshev Series 6875
Persistent Discretization Error in the Solution of Difference Equations 6877
—see also Codes, Debugging, Failures, Redundancy

Evaluation
Neutron Cross Section Data Evaluation, Storage and Retrieval 6881

F

Failures; Faults
Organization of Checking Experiments for Synchronous Sequential Machines 6828
Optimum Methods for Fault Diagnosis and Self-Repair 6833
Minimization of Page Faulting in Paged Memory Systems 6850
—see also Diagnosis, Reliability

Files; Filing
Automatic Filing of Manuscripts and Programs on LINC Tapes 6847
Integrity of a Mass Storage Filing System for Multiaccess 6851
Manipulation and Editing of Formatted Data Files 6852
—see also Sorting

Films
—see Magnetic Films

Formal
Formal Methods of Diagnosis in Engineering and Medicine 6866
—see also Languages

Format
Manipulation and Editing of Formatted Data Files 6852
Text-Processing Program for Editing, Formatting and Printing Finished Documents 6861
Structure and Generation of Display Formats for Interactive Problem Solving 6867

Fourier
Generation of Bit-Reversed Addresses for the Fast Fourier Transform 6831

Functions
State Transition Functions of Maximum Automata and Membership Functions of Fuzzy Sets 6827
Objective Function for Computer Systems Scheduling 6849
Hierarchical Method of Designing Statistical Recognition Functions 6862
Distance Functions on Digital Pictures 6866
Equivalence of Two Algorithms for Cooper’s Generalized Factorial Function 6869
—see also Approximation, Boolean, Threshold

G

Gates
Two-Level Realization of Switching Functions with a Minimum Number of Threshold Gates 6815
Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming 6816

Search for Tree Networks with Reduced Gate Count or Levels of Logic 6818
Allocation of Logic Gates to Modules for Minimum Delay 6819
Minimization of Gates and Connections in Combinational Logic Networks 6821

Generation
Generation of Bit-Reversed Addresses for the Fast Fourier Transform 6831
Programming Techniques for Generating Half-Tone Images for Optical Signal Processing and Data Portrayal 6860
—see also Displays, Noise, Numbers

Geometry
Use of Graph Theory and Geometry in the Solution of Simultaneous Linear Equations 6873
Graph Theory and Geometrical Techniques for Use in Contour Map Processing 6885
Geometric Relationships for Retrieval of Geographic Information Related to City Maps 6888

Graph Theory; Graphs
Graph Decomposition for Optimal Logic Module Clustering 6819
Use of Graph Theory and Geometry in the Solution of Simultaneous Linear Equations 6873
Graph Theory and Geometrical Techniques for Use in Contour Map Processing 6885
—see also Trees

Graphics
Auxiliary-Storage Associative Graphics Data Structure for PL/I 6834
Problem-Oriented Language for Handling Three-Dimensional Geometric Information 6855
Package of General Purpose Graphic Subroutines for Interactive Displays Using FORTRAN 6856
Debugging of Graphics Applications Programs Using DISPLAYTRAN 6858
Data-Structure for Graphic Display of Three-Dimensional Objects 5869
System for Interactive Problem Solving Using a Graphic Display Console 6867
Use of Graphics in a Heuristic Approach to Problem Solving 6882
Crystal Structure Analysis Using an Interactive Graphic Console 6882
—see also Displays, Images, Plotting

Heuristic
Solution of Word Sum Puzzles by a Heuristic Tree Searching Method 6868
—see also Problem Solving

Hierarchical
Hierarchical Method of Designing Statistical Recognition Functions 6862

Images
Programming Techniques for Generating Half-Tone Images for Optical Signal Processing and Data Portrayal 6960
—see also Graphics

Inequalities
Solution of a Set of Simultaneous Inequalities with Integer-Valued Variable 6868

Information
Problem-Oriented Language for Handling Three-Dimensional Geometric Information 6853
—see also Codes, Data, Retrieval
ABSTRACTS OF CURRENT COMPUTER LITERATURE

Input-Output
Simultaneous Input-Output and Instruction Processing 6832
Operating System Interfaces for High Level Languages Using Basic Input-Output Support Facilities 6848
—see also Consoles, Displays, Man-Machine Communications, Terminals

Instructions
Mathematical Model of Overlapping Memory References with Instruction Execution 6832

Integrated Circuits
Integrated Circuit Technique for Saturate Control of Switching Transistors 6834
Simulation of an Integrated Circuit IGFET Storage Cell 6835
—see also Arrays, Cells

Integration
Use of High Order Implicit Integration Algorithms for Transient Solution of Nonlinear Switching Circuits 6820
Numerical Integration of Conservative Systems of First-Order Ordinary Differential Equations 6876
—see also Differential Equations

Interaction
—see Man-Machine Communications

Interconnections
Data Communication Control Procedures for USA Standard Code for Information Interchange 6814

Interconnections
Module Clustering to Minimize Delay in Digital Network Interconnections 6819
Minimization of Gates and Connections in Combinational Logic Networks 6821
—see also Integrated Circuits, Interfaces

Interfaces
Operating System Interfaces for High Level Languages Using Basic Input-Output Support Facilities 6848
Interface Between a Conversational Job Processor and Its Operating System 6857

Interpreters
System for Interprettely Executing FORTRAN Statements Entered at a Display Console 6858

Languages
Recursively Unsolvable Problems on Context-Free Languages with Application to On-Line Language Recognizers 6822
Determining Time and Tape Complexity of Pushdown Automaton Languages 6823
Languages of String-Form Encoded Line Patterns 6865
Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition 6865
—see also Context-Free, Programming Languages, Sets

Learning
Generalized Portrait Method of Learning Applied to Diagnostics 6866
—see also Perceptrons

Linear Programming
Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming 6816

Lists
Definition and Implementation of LSIX Low Level Linked List Language in BCL 6845
On-List Line Assembly Program for 2048-Word LINC Computer 6847

Logic
Allocation of Logic Gates to Modules for Minimum Delay 6819
—see also Arrays, Cells, Combinational, Logic Design, Threshold

Logic Design
Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming 6816
Computer Simulation of Digital Design Logic 6817
Logic Design Automation of Fan-In Limited NAND Networks 6818
Logic Design Automation 6821

M
Machines
—see Automata, Computers, Man-Machine Communication, Sequential Machines

Magnetic Films
Improving the Signal-to-Noise Ratio of Photon and Electron Beam Accessed Magnetic Film Storage Arrays 6840
Feasibility of Lorentz Readout of a High-Density Fast Magnetic-Film Store 6841

Magnetic Recording
Computer Simulation of Unbiased Digital Recording 6842

Magnetic Tapes
File Management for a Fixed Disc Store Backed up by Magnetic Tape 6851

Magnetic Wires
High-Speed dc Coupled Digit Detector Array for Plated-Wire Storage Systems 6837
Plated Wire 200-ns Main Store 6838
80-ns Plated-Wire Store for a Time-Compression Multiplex Transmission System 6839

Management
File Management for a Fixed Disc Store Backed up by Magnetic Tape 6851
—see also Computer Center Management, Data Management, Files

Man-Machine Communications
—see Consoles, Displays, Terminals

Mathematical Programming
—see Linear Programming, Optimal

Mathematics
—see Arithmetic, Boolean, Differential Equations, Fourier, Geometry, Graph Theory, Integration, Models, Numerical Analysis, Optimal, Polynomials, Series, Statistics, Theorem Proving

Matrices
Use of Sparse Matrix Methods for the Transient Solution of Nonlinear Switching Circuits 6820
Organizing Matrices and Matrix Operations for Paged Memory Systems 6850
—see also Arrays

Measurement
Dichotomous Classification of a Binary Measurement Space 6862

Memory
—see Storage

Microelectronics
—see Integrated Circuits

Minimal; Minimization
Module Clustering to Minimize Delay in Digital Network Interconnections 6819
Minimization of Gates and Connections in Combinational Logic Networks 6821
Minimization of Page Faulting in Paged Memory Systems 6850
Recognition Functions Which Satisfy Both Least Square Error and Minimum Decision Error Rate Properties 6862
—see also Optimal

Models
Mathematical Model of Overlapping Memory References with Instruction Execution 6832
Modeling and Simulation of Insulated-Gate Field-Effect Transistor Switching Circuits 6835
Modeling in Three Dimensions 6859
—see also Simulation

Modules
Module Clustering to Minimize Delay in Digital Network Interconnections 6819
Analysis of Processor "Look-Ahead" and Simultaneous Operation of a Multi-Module Main Store 6832
Control of the Execution of Problem Solving Applications Modules on a Graphic Console 6867

Multi-Access
Integrity of a Mass Storage Filing System for Multi-Access 6851
—see also Time-Sharing

Multiplexing
80-ns Plated-Wire Store for a Time-Compression Multiplex Transmission System 6839

Multiplication
Binary Multiplication with Overlapped Addition Cycles 6830

N
Networks
Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming 6816
Logic Design Automation of Fan-In Limited NAND Networks 6818
Module Clustering to Minimize Delay in Digital Network Interconnections 6819
Acyclic Combinational Networks Composed of Primitive Blocks 6819
Minimization of Gates and Connections in Combinational Logic Networks 6821
Methods of Computing Event Times in Project Networks 6887
—see also Arrays, Modules, Switching

Noise
Improving the Signal-to-Noise Ratio of Photon and Electron Beam Accessed Magnetic Film Storage Arrays 6840
Local Smoothing of Noisy Data by Polynomials in n Dimensions 6871

Numbers
Subtraction of Negative Radix Numbers 6829

Numerical Analysis
Efficient Numerical Analysis of Nonlinear Switching Circuits 6820
Polynomial Curve Fitting When Abscissas and Ordinates are Both Subject to Error 6870
Local Smoothing of Noisy Data by Polynomials in n Dimensions 6871
—see also Arithmetic, Differential Equations, Differentiation, Functions, Integration, Matrices

On-Line
Recursively Unsolvable Problems on Context-Free Languages with Application to On-Line Language Recognizers 6822
On-Line List Assembly Program for 2048-Word LINC Computer 6847
On-Line Interactive Analysis and Display of High-Energy Physics Data 6880

Operating Systems
Operating System Which Allows a Program to Write in FORTRAN, ALGOL or JOVIAL 6846
Operating System Interfaces for High Level Languages Using Basic Input-Output Support Facilities 6848
Interface Between a Conversational Job Processor and Its Operating System 6857
Relationships Among the Operating System, the Display Terminal and the Computing System in DISPLAYTRAN 6868

Optical
Programming Techniques for Generating Half-Tone Images for Optical Signal Processing and Data Portrayal 6860

Optimal
Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming 6816
Optimum Methods for Fault Diagnosis and Self-Repair 6833
Orthogonalization of Search Vectors in Direct Search Optimization Methods 6879
—see also Minimal Output
—see also Input-Output

P
Pages; Paging
Minimization of Page Faulting in Paged Memory Systems 6830
Set-Theoretic Data Structure Suitable for Paging in a Mass Memory System 6853
Parallel
Propagating Adder Control Signals in a Parallel Binary Multiplier 6830
Patterns
Feature Extraction in Pattern Recognition by Kernel Approximation 6863
Recursive Bayesian Estimation Procedure for Pattern Recognition 6864
Perceptron Training Algorithm for Pattern Classification 6864
Application of Linguistic and Automaton Theory to Pattern Analysis and Recognition 6865
—see also Characters, Perceptrons
Perceptrons
Perceptron-Like Training Algorithm for Pattern Classification 6864

Peripheral Equipment
—see Consoles, Displays, Input-Output, Plotters, Storage, Terminals
Plotting
Plotters
Programs for Digitally Plotting any View of a Three-Dimensional Object 5869
Image Generation Using Digital Computers and Microfilm Plotters 6860
Polynomials
Polynomial Curve Fitting When Abscissas and Ordinates are Both Subject to Error 6870
Local Smoothing of Noisy Data by Polynomials in n Dimensions 6871
Problems Solving
System for Interactive Problem Solving Using a Graphic Display Console 6867
Use of Graphics in a Heuristic Approach to Problem Solving 6882
—see also Heuristic, Theorem Proving
Processing; Processors
Analysis of Processor "Look-Ahead" and Simultaneous Operation of a Multi-Module Neutron Module Test Facility 6812
Methods for Programs to Communicate with Command Language Processors 6848
Conversational Graphic Processor Using a Display Console 6857
Programming Techniques for Generating Half-Tone Images for Optical Signal Processing and Data Portrayal 6860
—see also Associative, Computers, Information, Lists, Parallel
Programmable
Automatic Programming Techniques 6846
Programmable Techniques for Generating Half-Tone Images for Optical Signal Processing and Data Portrayal 6860
—see also Heuristic
Programming Languages
Syntactic Extensions to the Functional Programming Language ISWIM 6844
Definition and Implementation of LSIX Low Level Linked List Language in BCL 6845
General Purpose Programming Language with Special Emphasis on Data Structures 6845
Operating System Interfaces for High Level Languages Using Basic Input-Output Support Facilities 6848
Methods for Programs to Communicate with Command Language Processors 6848
Auxiliary-Storage Associative Graphics Data Structure for PL/I 6854
Problem-Oriented Language for Handling Three-Dimensional Geometric Information 6855
Command Language, FORTRAN IV Subset and Graphics Language Provided for DISPLAYTRAN Users 6858
—see also Compilers, Languages
Programs; Routines
Proving Theorems About Programs Using Structural Induction 6844
On-Line List Assembly Program for 2048-Word LINC Computer 6847
Methods for Programs to Communicate with Command Language Processors 6848
Package of General Purpose Graphic Subroutines for Interactive Displays Using FORTRAN 6856
Debugging of Graphics Applications Programs Using DISPLAYTRAN 6858
Generalized Data Structure and Programs for Interactive Problem Solving Using a Graphic Display Console 6867
Compiler and Language for Chemical Plant Design and Simulation Programs 6883
—see also Debugging, Paging
Programs for:
Designing Optimal Networks with nor Gates for all Boolean Functions of Three Variables 6816
Computer Simulation of Digital Design Logic 6817
Logic Design Automation of Fan-In Limited NAND Networks 6818
Manipulation and Editing of Formatted Data Files 6852
Modeling in Three Dimensions 6859
Text-Processing 6861

Solution of Word Sum Puzzles by a Heuristic Tree Searching Method 6868
On-Line Interactive Analysis and Display of High-Energy Physics Data 6880
Neutron Cross-Section Data Evaluation, Storage and Retrieval 6881
Cam Design on an Interactive Graphic Display Console 6884

Proof
Proof of a Tree Sorting Algorithm 6844
—see also Theorem Proving

R
Realization; Implementation
Two-Level Realization of Switching Functions with a Minimum Number of Threshold Gates 6815
Definition and Implementation of LSIX Low Level Linked List Language in BCL 6845
—see also Design, Synthesis
Recognition
Recursively Unsolvable Problems on Context-Free Languages with Application to On-Line Language Recognizers 6822
Hierarchical Method of Designing Statistical Recognition Functions 6862
—see also Characters, Patterns, Recursive
Recursively Unsolvable Problems on Context-Free Languages with Application to On-Line Language Recognizers 6822
Recursive Bayesian Estimation Procedure for Pattern Recognition 6864
Reduction
Computer Reduction of Two-Level, Multiple-Output Switching Circuits 6821
—see also Minimal
Redundancy
Configurational and Functional Redundancy in Self-Repairing Computers 6833
—see also Multiplexing
Reliability
Optimally Reliable Computing Systems 6833
—see also Diagnosis, Error, Failures, Redundancy, Repair
Remote
—see Consoles, Terminals, Time-Sharing
Repairing
Analysis and Synthesis of a Self-Repairing Computer 6833

Retrieval
Neutron Cross-Section Data Evaluation, Storage and Retrieval 6881
Geometric Relationships for Retrieval of Geographic Information Related to City Maps 6888
—see also Search

S
Scheduling
Optimal Function for Computer Systems Scheduling 6849
Search
Search for Tree Networks with Reduced Gate Count or Levels of Logic 6818
Solution of Word Sum Puzzles by a Heuristic Tree Searching Method 6868
Orthogonalization of Search Vectors in Direct Search Optimization Methods 6879
—see also Files, Retrieval
Self-Repair

—see Repairing

Sequences
Sets of Sequences Which Synchronize Automata 6826
Checking Sequences for Synchronous Sequential Machines 6828

Sequential Machines
Organization of Checking Experiments for Synchronous Sequential Machines 6828
—see also Automata

Series
Error Estimation for the Solution of Ordinary Differential Equations in Chebyshev Series 6875

Sets
Sets of Sequences Which Synchronize Automata 6826
State Transition Functions of Maximum Automata and Membership Functions of Fuzzy Sets 6827
Set-Theoretic Data Structure Suitable for Paging in a Mass Memory System 6853

Signals
Propagating Adder Control Signals in a Parallel Binary Multiplier 6830
Improving the Signal-to-Noise Ratio of Photon and Electron Beam Accessed Magnetic Film Storage Arrays 6840
Programming Techniques for Generating Halftone Images for Optical Signal Processing and Data Portrayal 6860
—see also Fourier

Simulation
Computer Simulation of Digital Design
Logic 6817
Modeling and Simulation of Insulated-Gate Field-Effect Transistor Switching Circuits 6835
Computer Simulation of Unbiased Digital Recording 6842
Simulation of Degrees of Surface Transparency 6859
Compiler and Language for Chemical Plant Design and Simulation Programs 6883
—see also Models

Software
—see Operating Systems Programming Programs

Sorting
Proof of a Tree Sorting Algorithm 6844

Standards
Data Communication Control Procedures for USA Standard Code for Information Interchange 6814

Statements
System for Interpretively Executing FORTRAN Statements Entered at a Display Console 6858

States
State Transition Functions of Maximum Automata and Membership Functions of Fuzzy Sets 6827
—see also Automata Sequential Machines

Statistics; Statistical
Hierarchical Method of Designing Statistical Recognition Functions 6862
Recursive Bayesian Estimation Procedure for Pattern Recognition 6864

Storage
Computer-Aided Design of an 18-Bit 14K Computer Store 6817
Analysis of Processor "Look-Ahead" and Simultaneously Operation of a Multi-Module Main Store 6832
Simulation of an Integrated Circuit IGFET Storage Cell 6835
Switching Characteristics of Garnets Required for Their Use as Storage Elements 6836
Integrity of a Mass Storage Filing System for Multi-Access 6851
Auxiliary-Storage Associative Graphics Data Structure for PL/I 6854
Neutron Cross Section Data Evaluation, Storage and Retrieval 6881
—see also Allocation, Arrays, Associative, Electron Beams, Files, Integrated Circuits, Magnetic Films, Magnetic Tapes, Magnetic Wires, Paging, Retrieval

Strings
Program for Preparing and Editing Character Strings 6847
Languages of String-Form Encoded Line Patterns 6865

Subtraction
Subtraction by Minuend Complementation in Additive Systems 6829
—see also Addition

Switching
Efficient Numerical Analysis of Nonlinear Switching Circuits 6820
Computer Reduction of Two-Level Multiple-Output Switching Circuits 6821
Integrated Circuit Technique for Saturating Control of Switching Transistors 6834
Modeling and Simulation of Insulated-Gate Field-Effect Transistor Switching Circuits 6835
Switching Characteristics of Garnets Required for Their Use as Storage Elements 6836
—see also Automata, Logic, Networks

Switching Functions
—see Automata, Boolean, Logic, Sequential Machines, Threshold

Synthesis
Analysis and Synthesis of a Self-Repairing Computer 6833
—see also Design, Realization

Systems
Computer-Aided Data Dictionary for Systems Analysis Documentation 6890
—see also Communication, Computer Systems, Control, Information, On-Line, Operating Systems, Time-Sharing

Terminals
Relationships Among the Operating System, the Display Terminal and the Computer System in DISPLAYTRAN 6858
—see also Consoles, Display, Input-Output

Ternary
Designing Optimal Networks with NOR Gates for all Boolean Functions of Three Variables 6816

Tests
—see Debugging, Diagnosis, Evaluation

Textual Data Processing
—see Documentation

Theorem Proving
Proving Theorems About Programs Using Structural Induction 6844

Threshold
Two-Level Realization of Switching Functions with a Minimum Number of Threshold Gates 6815
Logical Design of an Optimal Network of Threshold Gates by Integer Linear Programming 6816

Time-Sharing
Revenue Analysis of Time-Sharing Computer Installations 6849
—see also Multiple-Access

Training
Perception-Like Training Algorithm for Pattern Classification 6864
—see also Learning

Transistor Circuits
Modeling and Simulation of Insulated-Gate Field-Effect Transistor Switching Circuits 6835
—see also Integrated Circuits

Transmission
80-ns Plated-Wire Store for a Time-Compression Multiplex Transmission System 6839
—see also Communications

Trees
Search for Tree Networks with Reduced Gate Count or Levels of Logic 6818
Proof of a Tree Sorting Algorithm 6844
Solution of Word Sum Puzzles by a Heuristic Tree Searching Method 6868
—see also Strings
IDENTIFIER INDEX

A
AC Flip-Flops 6817
Acceptance Powers 6824
Acyclic Networks 6819
ALGOL Compiler 6883
ALGOL Programming 6846
AND Gates 6816
Animation of Displays 6856
ASA FORTRAN Program 6852
ATLAS Computer 6845
Auxiliary Storage 6854

B
Balanced Word Line Driving 6838
Bayesian Estimation 6864
BCL Systems 6829
BCL Language 6845
Biharmonic Problems 6878
Bioinstrumentation 6847
Bit Density 6841
Bit Reversal 6831
Bit Sequencing 6814
Blasius Equation 6872
Boolean Functions 6815
Boundary Conditions 6877
Boundary Techniques 6878
Brodyen's Method 6874
Business Applications 6889

C
Cam Design 6884
Capitalization 6861
Cascade of Machines 6826
Catenation 6824
Cavity Flow 6878
CDC 1604 B Object Code 6846
Chain Code 6865
Checking Sequences 6828
Closure 6824
Cluster Detection 6866
Clustering of Modules 6819
Coates-Lewis Algorithm 6815
Coefficient Matrices 6873
Command Language 6858
Command Language Processors 6848
Common Mode 6837
Complementation 6829
Computer-Aided Analysis 6835
Computer-Aided Data Dictionary 6890
Computer-Aided Data Libraries 6882
Computer-Aided Design 6817, 6820
CONCOMP Project 6853
Concurrent ADD Cycles 6830
Configurational Redundancy 6833
Congruence Relations 6825
Conjunction 6824
Connections 6821
Conservative Systems 6876
Constraint Function 6815
Contour Maps 6885
Conversational Display 6857, 6858
Cooper's Function 6869
Crank-Nicolson Method 6877
Cryptarithm Puzzles 6868
Crystal Structure Analysis 6882
Curie Point Writing 6841
Curve Fitting 6870, 6871
Curve Plotting 6860
CYTOS 6848

D
Davidenko's Method 6874
Definite Automata 6826
Deming's Method 6870
Detectors (Digit) 6837
Dichotomy Problem 6862
Dictionaries 6890
Deterministic Automata 6823, 6827
Difference Equations 6877
Digit Detectors 6837, 6838, 6839
Digit Driving 6838
Discretization Error 6877
Disjunction 6824
DISPLAYTRAN System 6858
Dissection of Pictures 6866
Distance Functions 6866
Distinguishing of Automata 6826
Dumping of Files 6851
Efficiency of Analysis 6820
Embedded Facilities 6848
Encryption of Images 6860
Engineering Diagnosis 6886
Entropy Minimization 6863
Equivalent Circuits 6835
Equivocility Curves 6878
Establishment/Termination Procedures 6814
Estimation Model 6864
Estimation of Errors 6875
Expected Decision Error 6862
Expression Compiler 6844
Factorial Function 6869
Factoring 6818
Fan-In Limit 6818
Fast Fourier Transform 6831
Feasibility Study 6841
Feature Extraction 6863
Field Handling Routines 6845
Field-Effect Transistors 6835
Finite Automata 6826
Finite Difference Method 6878
Flip-Flop 6817
Flow in Cavities 6878
FORTRAN Program 6861
FORTRAN Programming 6846
FORTRAN Programs 6818
FORTRAN Statements 6858
FORTRAN Subroutines 6856
FORTRAN IV Programs 6861
FORTRAN-Like Language 6896
Freeman Chain Code 6865
Frequency Dictionary 6861
Functional Programming Language 6844
Functional Redundancy 6833
Fuzzy Sets 6827

G
Garnets 6836
Generalized Congruence 6825
Generalized Isomorphism 6825
Generalized Portrait Method 6886
Geographic Information 6888
Half Tone Images 6860
Heat Conduction Equation 6877
Hidden Line Problem 6859
High-Density Store 6841
High-Energy Physics Data 6880
High-Speed Memories 6837, 6838, 6841
Homing of Automata 6826
Homomorphisms 6825
IBM 1108 Applications 6852
IGFET Memory Cell 6835
Implicit Integration 6820
Incidence Matrices 6873
Inconsistency Removal 6815
Incrementing Bit-Reversed Integers 6831
Indexing 6831
Input Sequences 6826
Integer Incrementing 6831
Integer Linear Programming 6816
Integrity of Storage Systems 6851
Interlaced Memories 6832
Isomorphism 6825
ISWIM Language 6844

H
J
JK Flip-Flops 6817
Job Control 6857
JOVIAL Programming 6846
Justification of Text 6861
k-Definite Automata 6826
Kernel Approximation 6863

L
Lagrangian Multiplier 6862
Landin's Language 6844
LAP 6 Program 6847
Least-Squares 6862, 6870, 6871
LEGER Program 6859
LINC Computer 6847
Line Patterns 6865
Linear-Bounded Automaton 6824
Links (Communication) 6814
Loading of Programs 6847
Loading Statistics 6817
Look-Ahead of Processors 6832
Lorenz Readout 6841
LSIX Language 6845

M
Manuscript Editing 6847
Maps (City) 6888
Maps (Contour) 6885
Mass Storage 6851, 6853
Maximin Automata 6827
Medical Diagnosis 6886
Membership Functions 6827
Message Transfer 6814
Meta Commands 6847
Microfilm Plotters 6860
Minimax Approximation 6872
Minuend Complementation 6829
Modification of Displays 6856
Moiré Pattern Generation 6860
Monolithic Circuits 6835
Multi-Module Stores 6832
Multiple-Output Switching Functions 6821
Multitape Automata 6824
ABSTRACTS OF CURRENT COMPUTER LITERATURE

N
NAND Gates 6816, 6817, 6818
Navier-Stokes Equation 6878
Navigation 6885
Negative Radix Numbers 6829
Nondeterministic Automata 6823, 6827
Nonlinear Approximation 6872
Nonlinear Circuits 6820
Nonlinear Equations 6874
Nonwriting Automata 6824
NOR Gates 6815, 6819

O
Object Code 6816
Objective Functions 6819, 4869
One Shots 6817
OR Gates 6816
Ordering 6887
Orthogonalization 6879
Overlapped Addition 6830
Overlapped Memory Access 6832

P
Package of Subroutines 6856
Packaging of Networks 6819
Partitioned Matrix 6850
Partitioning of Networks 6819
Permutation Matrices 6873
Permutation-Reset Machines 6826
Persistent Discretization Error 6877
PERT Event Times 6887
Physics Data 6880
Pictures 6866
PL/I Data Structure 6854
Plated Wire Memories 6837, 6838
Polling/Selection Procedures 6814
Polyhedral Approximation 6859
Predicates 6824
Primitive Blocks 6819
Probabilistic Automata 6827
Project Networks 6887
Projection Methods 6873
Propagating Adder Control Signals 6830
Propagation Time 6817
Pseudofiles 6848
Pushdown Automata 6823
Puzzles 6868
Random Access Store 6841
Recursion Induction 6844
REFORM Program 6852
Relational Homomorphisms 6825
Replication 6819
Reset Machines 6826
Residual Computation 6874
Revenue Functions 6849
Reversed Integer 6831
Reynolds Number 6878
Rosenbrock-Swann Method 6879
Round-Off Errors 6876
Saturation Control of Transistors 6834
Schematics 6817
SCORE Program 6881
Separable Equations 6877
Separated Boundary Conditions 6877
Serial Read 6839
Serial Write 6839
SIGHT Program 6859
Smoothing of Data 6871, 6878
Sparse Matrices 6820, 6873
Stacks (Memory) 6838
STDS 6853
Storage Time in Transistors 6834
Structural Induction 6844
Successive Approximation 6873
SUMX Programs 6880
Surface Transparency 5869

S
Synchronization of Automata 6826
Synchronous Adder 6830
Synchronous Sequential Machines 6828
System-Effectiveness Measure 6833

T
Tape Complexity 6823
Time Complexity 6823
Timing Diagram 6817
Topological Ordering 6887
Topological Properties 6885
Transients 6820
Transition Checking 6828
Transition Functions 6827
Transitive Closure 6824
Triples 6854
Truncation Error 6875
Turing Machines 6823, 6824
Two-Level Switching Circuits 6815, 6821
Two-Rail Logic 6837
Two-Tier File Dumping 6851
Two-Way Automata 6824

U
Ultimate-Definite Automata 6826
UNIVAC 1218 Object Code 6846
Unsolvable Problems 6822
USASCII 6814

V
Virtual Memory Systems 6850
Vocabulary 6861
Vortices 6878

W
Walsh Function 6862
Word Sum Puzzles 6868
Woven Wire Memories 6838
WRITE Transient 6837

Z
Zeiger's Procedure 6826
## AUTHOR INDEX

<table>
<thead>
<tr>
<th>A</th>
<th>Aho, A. V. 6823</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Appel, A. 6859</td>
</tr>
<tr>
<td>B</td>
<td>Beisner, H. M. 6864</td>
</tr>
<tr>
<td></td>
<td>Berns, G. M. 6861</td>
</tr>
<tr>
<td></td>
<td>Bielowski, H. 6849</td>
</tr>
<tr>
<td></td>
<td>Brown, S. H. 6857</td>
</tr>
<tr>
<td></td>
<td>Broyden, C. G. 6874</td>
</tr>
<tr>
<td></td>
<td>Buckle, D. G. R. 7889</td>
</tr>
<tr>
<td></td>
<td>Burstall, R. M. 6844, 6868</td>
</tr>
<tr>
<td>C</td>
<td>Calahan, D. A. 6820</td>
</tr>
<tr>
<td></td>
<td>Cameron, P. T. 6883</td>
</tr>
<tr>
<td></td>
<td>Capocaccia, F. 6834</td>
</tr>
<tr>
<td></td>
<td>Carroll, B. D. 6815</td>
</tr>
<tr>
<td></td>
<td>Chen, F. C. 6867</td>
</tr>
<tr>
<td></td>
<td>Childs, D. L. 6853</td>
</tr>
<tr>
<td></td>
<td>Coates, C. L. 6815</td>
</tr>
<tr>
<td></td>
<td>Coffman, Jr., E. G. 6850</td>
</tr>
<tr>
<td></td>
<td>Cohen, M. S. 6841</td>
</tr>
<tr>
<td></td>
<td>Cole, A. J. 6871</td>
</tr>
<tr>
<td></td>
<td>Comba, P. G. 6855</td>
</tr>
<tr>
<td></td>
<td>Cornish, R. E. 6858</td>
</tr>
<tr>
<td></td>
<td>Creasy, R. J. 6881</td>
</tr>
<tr>
<td></td>
<td>Cutlip, W. F. 6826</td>
</tr>
<tr>
<td>D</td>
<td>Davies, A. J. T. 6871</td>
</tr>
<tr>
<td></td>
<td>Day, C. W. 6843</td>
</tr>
<tr>
<td></td>
<td>Dietmeyer, D. L. 6818, 6821</td>
</tr>
<tr>
<td></td>
<td>Dorrough, D. B. 6833</td>
</tr>
<tr>
<td></td>
<td>Dougherty, R. L. 6867</td>
</tr>
<tr>
<td>F</td>
<td>Feder, J. 6865</td>
</tr>
<tr>
<td></td>
<td>Fenwick, P. M. 6830</td>
</tr>
<tr>
<td></td>
<td>Fraser, A. G. 6851</td>
</tr>
<tr>
<td></td>
<td>Frater, D. 6831</td>
</tr>
<tr>
<td>G</td>
<td>Gagliano, F. W. 6858</td>
</tr>
<tr>
<td></td>
<td>Gilbert, P. 6846</td>
</tr>
<tr>
<td></td>
<td>Gonenc, G. 6828</td>
</tr>
<tr>
<td></td>
<td>Greenspan, D. 6878</td>
</tr>
<tr>
<td></td>
<td>Gunn, D. M. 6846</td>
</tr>
<tr>
<td></td>
<td>Gupta, S. C. 6832</td>
</tr>
<tr>
<td>H</td>
<td>Hays, G. G. 6817</td>
</tr>
<tr>
<td></td>
<td>Heightley, J. D. 6839</td>
</tr>
<tr>
<td></td>
<td>Hilsenrath, J. 6852</td>
</tr>
<tr>
<td></td>
<td>Hodges, D. A. 6835</td>
</tr>
<tr>
<td></td>
<td>Howry, S. K. 6880</td>
</tr>
<tr>
<td></td>
<td>Housden, R. J. W. 6845</td>
</tr>
<tr>
<td></td>
<td>Hunt, C. H. 6889</td>
</tr>
<tr>
<td>I</td>
<td>Ibaraki, T. 6816</td>
</tr>
<tr>
<td></td>
<td>Ishii, O. 6838</td>
</tr>
<tr>
<td></td>
<td>Ito, T. 6862</td>
</tr>
<tr>
<td>J</td>
<td>Jacobsen, J. D. 6888</td>
</tr>
<tr>
<td></td>
<td>Jones, N. D. 6824</td>
</tr>
<tr>
<td>K</td>
<td>Kilmer, W. L. 6826</td>
</tr>
<tr>
<td></td>
<td>King, P. J. H. 6890</td>
</tr>
<tr>
<td></td>
<td>Kobayashi, K. 6822</td>
</tr>
<tr>
<td>L</td>
<td>Lafuente, J. M. 6884</td>
</tr>
<tr>
<td></td>
<td>Langdon, Jr. G. G. 6829</td>
</tr>
<tr>
<td>M</td>
<td>Lawler, E. L. 6819</td>
</tr>
<tr>
<td></td>
<td>Lerner, A. J. 6886</td>
</tr>
<tr>
<td></td>
<td>Levitt, K. N. 6819</td>
</tr>
<tr>
<td></td>
<td>Lynes, D. J. 6837</td>
</tr>
<tr>
<td>N</td>
<td>Mallinson, J. C. 6842</td>
</tr>
<tr>
<td></td>
<td>McCann, M. J. 6876</td>
</tr>
<tr>
<td></td>
<td>McClenon, R. C. 6852</td>
</tr>
<tr>
<td></td>
<td>McGee, W. C. 6880</td>
</tr>
<tr>
<td></td>
<td>McKellar, A. C. 6850</td>
</tr>
<tr>
<td></td>
<td>Morse, S. P. 6885</td>
</tr>
<tr>
<td></td>
<td>Muroga, S. 6816</td>
</tr>
<tr>
<td>O</td>
<td>Nitta, M. 6838</td>
</tr>
<tr>
<td></td>
<td>Nudds, D. 6887</td>
</tr>
<tr>
<td>P</td>
<td>Okaya, Y. 6882</td>
</tr>
<tr>
<td></td>
<td>Oliver, J. 6875</td>
</tr>
<tr>
<td></td>
<td>O'Neill, M. 6870</td>
</tr>
<tr>
<td></td>
<td>Osborne, M. R. 6872, 6877</td>
</tr>
<tr>
<td>R</td>
<td>Palmer, J. R. 6879</td>
</tr>
<tr>
<td></td>
<td>Penafiel, H. R. 6880</td>
</tr>
<tr>
<td></td>
<td>Pernesi, A. J. 6839</td>
</tr>
<tr>
<td></td>
<td>Pfaltz, J. L. 6866</td>
</tr>
<tr>
<td>S</td>
<td>Rechard, O. W. 6869</td>
</tr>
<tr>
<td></td>
<td>Robertson, H. H. 6876</td>
</tr>
<tr>
<td></td>
<td>Rosenfeld, A. 6866</td>
</tr>
<tr>
<td></td>
<td>Rosin, R. F. 6848</td>
</tr>
<tr>
<td></td>
<td>Rully, A. D. 6856</td>
</tr>
<tr>
<td></td>
<td>Ryder, M. M. 6889</td>
</tr>
<tr>
<td>T</td>
<td>Takashima, M. 6838</td>
</tr>
<tr>
<td></td>
<td>Tewarson, R. P. 6873</td>
</tr>
<tr>
<td></td>
<td>Thombs, H. W. 6858</td>
</tr>
<tr>
<td></td>
<td>Tou, J. T. 6863</td>
</tr>
<tr>
<td></td>
<td>Turner, J. 6819</td>
</tr>
<tr>
<td>U</td>
<td>Ullman, J. D. 6823</td>
</tr>
<tr>
<td>W</td>
<td>Watson, G. A. 6872</td>
</tr>
<tr>
<td></td>
<td>Wilkes, M. A. 6847</td>
</tr>
<tr>
<td>Y</td>
<td>Yeh, R. T. 6825</td>
</tr>
<tr>
<td>Z</td>
<td>Zimmerman, L. L. 6843</td>
</tr>
</tbody>
</table>