To submit papers: Send two copies of a paper of interest to the computer field to IEEE Computer Society Publications Office, 5855 Naples Plaza, Suite 301, Long Beach, CA 90803.

Include a 50-100 word abstract, a list of index terms (four maximum), and a cover letter giving permission to enter the paper in the Repository (entry in the Repository does not constitute publication).

All submitted material should be unbound, unstapled, printed one side only in black on 8½" x 11" white paper. Material not conforming to the above requirements will not be accepted.

To order papers: State the R-number, listed before the author's last name, of each paper you order. Prices are 12 cents per page, plus $1.00 service charge for orders under 50 pages.

Microfiche copies are available for $2.50 for manuscripts under 50 pages, plus $2.50 for each additional 50 pages or fraction thereof.

All Repository items must be prepaid except for companies or institutions with established accounts. A $3.00 invoice charge is added to all non-prepaid orders. Make your check or money order payable to the IEEE Computer Society.


R78-102—Parhami, B., “An Introduction to the Geometry of Digital Pictures” (12 pp., Arya-Mehr University of Technology, Tehran, Iran)

A digital picture is defined as a set of points on the XY plane with integer coordinates, called grid points, which represent geometric patterns in computer memories. The geometry of digital pictures defines the relationship between the patterns and the corresponding digital pictures. This paper lays the foundation of digital picture geometry by presenting fundamental definitions and initial results, with emphasis placed on properties of straight line segments.

R78-103—Parhami, B., “Interconnection Redundancy for Reliability Enhancement in Digital Systems” (23 pp., Arya-Mehr University of Technology, Tehran, Iran)

Even though interconnections are considerably more reliable than other module-level components in a digital system, the unreliability of interconnections contributes heavily to the overall system unreliability when the module-level components are protected by conventional redundancy techniques. This is even more significant when a common bus is used for data communication between modules. This paper investigates bus redundancy in terms of the additional complexity introduced by the switching circuits and the resulting improvement in reliability.

R78-104—Maynard, J. L., “Can We Improve User Manuals?” (12 pp., Honeywell Information Systems, Phoenix, Arizona)

This paper is based on 15 years of research by the author, the former manager of programming publications for Xerox Data Systems. It describes a new, more effective style of user manual designed to solve the many problems users encounter with industry-standard user manuals. It also suggests a more realistic role for the writer to play in software development, and a more sensible approach to the selection and training of writers, to the organization of manuals, and to the way they are produced and distributed.


A mixed-mode sequential machine is defined as a representation in which both synchronous (clocked) and asynchronous (unclocked) state transitions are allowed. The intent of the representation is to allow both types of transitions in a single structure, thereby promoting single LSI device realizations. The approach is thus different from conventional design methods, in which asynchronous problems are treated separately with separate devices. This paper covers the problems of state reduction, state assignment, and realization, and presents a ROM implementation which allows both clocked and unclocked transitions to occur.

R78-106—Mehra, S.K., J.W. Wong, and J.C. Majithia, “A Comparative Study of Some Nonhomogeneous Two-Processor Organizations” (56 pp., University of Waterloo, Waterloo, Canada)

Applications such as transaction processing often require independent processing by several processors. To achieve a small response time, some form of parallel processor organization could be used. Many such organizations are possible. This paper describes and evaluates, through queueing analysis or simulation, four alternative organizations for two nonhomogeneous processors. It compares their performance under different workloads and indicates the possibility of switching between the alternate architectures in a dynamically reconfigurable system.

R78-107—Parhami, B., “An Approach to Hardware and Software Fault Tolerance Through the Design of Self-Checking and Fail-Safe Programs” (29 pp., Arya-
Mehr University of Technology, Tehran, Iran

This paper considers techniques for making programs self-checking or fail-safe with respect to certain hardware and software faults (e.g., those affecting only a single program statement) by introducing redundancy into the program design. The redundancy can be applied at various levels, starting from single statements (instructions) to complete program modules. Several examples of such techniques are discussed with respect to their effectiveness in detecting hardware and software malfunctions.


Hardware redundancy methods and their effectiveness in coping with hardware faults are surveyed and the corresponding mathematical reliability models are presented. It is shown that under certain very broad conditions, usually satisfied by actual systems, considerable improvement in reliability can be achieved through judicious use of known redundancy schemes. These techniques include component-level replication, static and adaptive voting arrangements, triple-modular redundancy, standby sparing, and hybrid redundancy. Includes bibliography on fault tolerant computing.

R78-109—Martin, T., “Development of High Order Realtime Programming Language PEARL in Germany” (38 pp., Kernforshungszentrum Karlsruhe GmbH, Karlsruhe, Germany)

In 1976 the definition of PEARL was frozen; in 1977 the official language descriptions appeared, full PEARL being available in the language, and basic PEARL the joint minimum subset of all present implementations. This paper the authors describe the progress of development of the language and lists references to the documents and literature involved.


Using Bauer’s approach to relative error propagation, methods are presented for performing various error analyses on numerical algorithms. A technique is developed which generates a system of equations employed by the error analyses, and which subroutines is an order of magnitude less time and storage than present techniques. Accurate analyses are attempted by working in the $\infty$-norm, which requires the solution of minimax problems. A heuristic approximation method is also described. These methods are compared with the 2-norm approximation methods of Miller. The error analyses provide alternative criteria by which algorithms that solve the same problem may be compared. Additionally, the analysis of a composite algorithm, which is made up of concatenated subalgorithms is given in terms of analyses done on its parts. Finally, by using a forward analysis, limited algorithm improvement is obtained by the location of sensitive operations, and the substitution of mathematically equivalent expressions.

R78-111—Derivisoglu, B., “Iterative State Reduction of Sequential Machines” (20 pp., University of Connecticut, Storrs)

An iterative method of state reduction fully enables a minimized sequential machine. The method is based upon a new concept of “relative prime classes,” which is an extension of the prime compatibility class concept. Theorems are given which state that a closed cover can always be found using relative prime classes and testing if (when) the cover is minimal. This paper presents an algorithm which makes it possible to break a large state reduction problem into a number of much smaller ones and establishes criteria to determine if the minimal cover has been found.

R78-112—Joliat, M. L., “On the Reduced Matrix Representation of LR(k) Parser Tables” (161 pp., University of Toronto)

The representation of LR(k) grammar parser tables is considered in a reduced matrix structure that preserves the error detection capability of the unreduced parser. New definitions of state compatibility enable the use of a so-called Remainder-LR(k) parser tables by a simple non-enumerative state reduction algorithm. A heuristic state assignment procedure, applied before parser matrix reduction, significantly improves two-way state and symbol reduction of the parser matrix. Implementation space requirements of the matrix parser are shown to be competitive with existing representations. Parsing speed comparisons show the reduced matrix parser to be one of the fastest practical parsers known. The formal generation of a reduced matrix lexical analyzer that is as efficient as a handwritten scanner further illustrates the viability of the reduced matrix structure.


This paper considers the computational realization of a recursive sine generator and proposes a technique for reducing the round-off noise in fixed-point organizations. It presents two new bidirectional sine algorithms and derives the noise formulas.

R78-114—Hartimo, I. and L. Ojala, “Properties of Recursive Sine Algorithms” (14 pp., Helsinki University of Technology, Finland)

The roundoff noise properties of fixed-point digital filters are studied by using recursive sine generators as examples. Three new methods to realize the direct form II of a digital filter are suggested and their computational properties are described.

R78-115—Hartimo, I. and L. Ojala, “A Reliable Sine Generator” (10 pp., Helsinki University of Technology, Finland)

This paper describes a hardware realization of a recursive sine wave generator. The construction is based on the so-called sum-algorithm developed by the authors. The generator is aimed at experimental testing of different hypotheses concerning the nature of round-off noise accumulation in recursive sine generators.

R78-116—Hartimo, I., “Digital Oscillators” (4 pp., Helsinki University of Technology, Finland)

Recursive computation can be used to generate sine waveforms. The waveforms are degraded by the round-off noise generated in rounding the results of multiplications. The properties of the round-off noise are usually analyzed using statistical methods. The waveform generator is, however, a case where the noisy statistical assumptions do not hold. This paper describes the so-called ‘locking’ phenomena. A measure of the locking sensitivity of a certain algorithm is given, as well as some results of computer simulations.

R78-117—Choy, D. M. and C. K. Wong, “Optimal $\omega-\beta$ Trees with Capacity Constraint” (42 pp., IBM Thomas J. Watson Research Center, Yorktown Heights, New York)

The authors consider a specific kind of binary tree with weighted edges. Each right edge has weight $\omega$ while each left edge has weight $\beta$. Furthermore, no path in the tree is allowed to contain $L$ or more consecutive $\omega$-edges, where $L>1$ is fixed. Given $\omega$, $\beta$, $L$ and the number of nodes $n$, an optimal tree is one which minimizes the total weighted path length. Algorithms for constructing an optimal tree as well as all optimal trees for given $\omega$, $\beta$, $L$ and $n$ are proposed and analyzed. Timing and storage requirements are also discussed.


This paper considers the accessing of batched requests in a linear storage medium. The batch size is assumed to be fixed and the access probabilities of individual records are to be known. For a given

Order by R-number.
Use the Repository order form on p. 105.
arrangement of records in the storage, the authors consider the problem of read/write head scheduling to minimize the expected access time for a batch, measured in terms of the distance traveled by the head. The first part of the paper proposes, analyzes, and compares several simple algorithms. It also discusses the effect of different record arrangements. The second part of the paper describes a family of algorithms called B-optimal rules. When $B = \infty$, an $\infty$-optimal rule is indeed optimal in the sense of minimizing expected distance traveled by the head per batch, when accessing an arbitrarily large number of records. The paper describes a procedure, based on the idea of "discrete dynamic programming," to calculate an $\infty$-optimal rule for any given record arrangement.

R78-119—Stockmeyer, L. J. and C. K. Wong, "On the Number of Comparisons to Find the Intersection of Two Relations" (38 pp., IBM Thomas J. Watson Research Center, Yorktown Heights, New York)

Given two finite sets of $k$-tuples whose component elements are drawn from an infinite totally ordered set, the problem of identifying the $k$-tuples which belong to both sets is considered. Attention is restricted to algorithms that perform pairwise comparisons on the component elements of the $k$-tuples. If the two sets have cardinalities $m$ and $n$ with $m < n$ it is shown that, in the worst case, $(m+n)\log_2 m + (m+n-1)k$ comparisons are sufficient and $\max((m+n)\log_2 m - 2.9m, (m+n-1)k)$ comparisons are necessary. Upper and lower bounds are also given for the number of comparisons required to recognize duplicate tuples in a sequence of tuples, and to determine the lexicographic order of a sequence of tuples. In all cases, the disparity between the upper and lower bounds is at most a factor of two asymptotically.

R78-120—Coppersmith, D., Lee, D. T., and C. K. Wong, "Isometries Between $L_1$ and $L_\infty$ Metrics in Coordinate Space" (IBM Thomas J. Watson Research Center, Yorktown Heights, New York)

In a 2-dimensional coordinate space, there exists an isometry between $L_1$ and $L_\infty$ metrics. Consequently, results on computational complexity for one metric can be transplanted to the other in a natural way. The question of extending the isometry to higher dimensional spaces immediately arises. In this note the authors give a negative answer to this question by showing that in $L_1$ metrics, the maximum number of nonoverlapping unit spheres one can pack in a sphere of radius 2 (hence $2^k$ times the volume of a unit sphere) is exactly $2^k$.


A procedure is described for generating a two-parameter orthogonal transformation matrix which reduces to the Fourier and Hadamard transformation matrices under special conditions. This generalized transformation matrix is particularly useful for multidimensional signal processing on a real-time basis, because it preserves a proper relationship in the transform domain.

R78-122—Brunening, J. T., "Inverses of Transfer Function Matrices" (38 pp., Baker University, Baldwin City, Kansas)

This paper investigates a general form for inverse systems of linear sequential circuits. The form for the transfer function matrix is a linear combination of matrix extensions of the adjoints of square submatrices of the transfer function matrix of the original circuit. Any inverse system can be written in the form presented here. This result is then utilized to develop an algorithm which generates the adjoint and yields the minimal delay. Finally, upper bounds on the minimal degree of all polynomial inverses of a matrix are established. The results of this paper are limited to transfer function matrices of full rank.

R78-123—Clark, D. W., "Measurements of Dynamic List Structure Use in Lisp" (31 pp., Xerox Palo Alto Research Center, Palo Alto, California)

This paper is an empirical study of how three large Lisp programs use their list structure during execution. Most list-cell references are due to the functions car and cdr, which are executed about equally often and greatly outnumber other primitive functions. Executions of cdr yield the atom NIL about 10 to 20 percent of the time, and nearby list cells most of the rest of the time. Executions of car yield atoms, small integers, and list cells in varying proportions in the three programs. Atom references by car tended to concentrate on a small number of atoms. The function rplacd increases static pointer locality, but rplaca is used idiosyncratically. Repeated reference to list cells is likely: over half of all references were to one of the ten most recently referenced cells. Linearization is the rearrangement of lists so that consecutive cdr's are adjacent in memory whenever possible. This property deteriorates slowly after a list structure is linearized. If all of a program's lists are linearized, page-faults are reduced slightly, but because of the high cost of a fault this small reduction has a large effect.


This report presents abstracts describing recently completed projects, summary information on the status of current projects, and explanations of new projects at the Information Engineering Laboratory of the Department of Computer Science at the University of Illinois at Urbana-Champaign.

R78-125—Campbell, R. H. and T. J. Miller, "A Path Pascal Language" (23 pp., Report No. UIUCDCCS-R-78-919, University of Illinois, Urbana, Illinois)

This paper describes an implementation of open path expressions in Pascal. The extended language is used to gain experience with path expressions in the design and construction of practical real-time tradeoff between the two algorithms. The extended language includes an encapsulation mechanism and a novel technique which associates path expressions with access right exportation.

R78-126—Michalski, R. S., "Designing Extended Decision Tree Tables and Optimal Decision Trees Using Decision Diagrams" (57 pp., Report No. UIUCDCS-R-78-988, University of Illinois, Urbana, Illinois)

The paper introduces the concept of a decision diagram and shows its application to designing extended decision tables and converting them to space or time optimal decision trees. Two algorithms for optimal (or suboptimal) space or time conversion are described using decision diagrams. These algorithms are basically decomposition algorithms, but by varying their degree one can obtain a spectrum of algorithms differing in the way they convert the programs. Efficiency and the degree of guarantee that the solution is optimal. When the algorithms do not guarantee the optimality, they give a measure of the maximum possible distance between the obtained and the optimal trees.

R78-127—Srihari, S. N. and M. J. Ohanesian, "An Efficient Algorithm for Determining Hadamard Sequency Vectors" (7 pp., Wayne State University, Detroit, Michigan)

An efficient algorithm for determining the sequency vector $S_a$ of a $2^n \times 2^n$ Hadamard matrix is developed. The method requires fewer computation steps than a previously known method.


A general planar model of multidimensional discrete spaces (a diagram) is described which can be used for geometrically representing binary, multiple-valued, discrete and variable-valued logic functions. It is essentially a multiple-valued extension of Marquand's binary diagram, with an additional feature of varying thickness of lines representing axes of variables.