The Repository, a collection of over 2000 technical papers and documents relating to computer science and engineering, is maintained by the Computer Society as a service to the information processing community. Some of the papers have been refereed; others have not.

If you have a paper of interest to the computer field, you are invited to submit two copies, including abstract and index terms, to Dr. Warren L. Semon, IEEE Computer Society Editor-in-Chief, Division of Systems and Information Sciences, 313 Link Hall, Syracuse University, Syracuse, NY 13210. Be sure to include a cover letter giving permission to enter the paper in the Repository. Entry in the Repository does not constitute publication.

Photocopy prices are $1.00 per page. Add $1.00 service charge to all orders under 50 pages. Microfiche copies are available for $2.50 for each manuscript under 50 pages, plus $2.50 for each additional 50 pages or fraction thereof.

Be sure to state the R number, listed before the author's last name, of each paper you order. All Repository items must be prepaid except for companies or institutions with established accounts. A $2.00 invoice charge will be added to all non-prepaid orders. Please make your check or money order payable to the IEEE Computer Society.

R76-103—Chow, We-Min and Willy W. Chiu, "An Analysis of Swapping Policies in Virtual Storage Systems" (31 pp., IBM Thomas J. Watson Research Center, Yorktown Heights, New York)

An important resource allocation mechanism in virtual storage operating systems is the maintenance of the multi-programming level in main storage, especially when some form of working-set storage management strategy is employed. Swapping of programs in and out of main memory occurs when sufficient storage becomes available and when total storage demand exceeds capacity. In this paper, we propose a class of swapping algorithms that couples storage management parameters with swapping decisions. An analytic model is developed and numerical results are presented to compare the performance of these algorithms.

R76-104—Mikami, Yukihiro, "The Average Access Time of a Dynamic Semiconductor Memory System" (16 pp., Bell-Northern Research, Ottawa, Canada)

An analysis and report of experimental results on the average access time of a dynamic semiconductor memory system that shows the average access time increment due to the internal refresh is proportional to the refresh frequency and to the square of the access time of the memory system.

R76-105—Wormald, E. G., "A Note on Synchronizer or Interlock Maloperation" (3 pp., Australian Telecommunications Commission, Sydney, Australia)

This note discusses a way of avoiding thermal-operation of synchronizers or interlocks subject to meta-stable action when handling signals from another, differently-clocked system. Several possible variations are described.

R76-106—Landis, DeWitt, "Multiple Response Resolution in Associative Systems" (20 pp., Los Angeles, California)

The selection of a single responding word out of several responses to an associative search is a major and frequent function of associative memories and processors. To select a responder, an associative memory must be able to generate a signal that is one for the desired word and zero for all others. This paper describes two families of circuits for doing this. The first family, called the preliminary circuits, is based on a generalization of a previously published circuit. The second family, called the optimum circuits, is derived from the first, and the circuits are optimum in the sense that they are the fastest circuits the techniques of this paper can give. An analysis of the settling time (in gate delays) required for the operation of the optimum circuits shows that, while they lack some of the symmetry of the preliminary circuits, they are the fastest practical circuits yet proposed for multiple response resolution.

R76-107—Sintonen, Leo, "A Clocked Multi-Valued Flip-Flop" (9 pp., Tampere University of Technology, Tampere, Finland)

A multi-valued flip-flop is presented. The implementation of the flip-flop is independent of N (number of logic levels). The output of the flip-flop is stable during clock transitions. Techniques in the design of sequential circuits are presented.


This paper investigates the application of Burst processing to the problem of tuning and demodulating FM signals using digital hardware. Such digital FM receivers are shown to be conceptually sound and capable of worthwhile trade-offs of performance and economy. These results provide the basis for the implementation of a new class of digital FM receiver. The relative performance of the different configurations of the Burst receiver is discussed.

R76-109—Salisbury, Alan B., "The Evaluation of Microprogram Implemented Emulators" (123 pp., Digital Systems Laboratory, Stanford University, Stanford, California)

A new technique for estimating the performance of microprogram implemented emulators is developed and presented. This technique of "synthetic emulation" combines the concepts of mixes, kernel programs, synthetic programs, and mathematical models, all adapted to the
microprogramming environment. Methods are presented for the application of this technique to the problem of selecting “host” microprocessors for the emulation of one or more “target” machines in a known workload environment. The process of emulation is examined and target machine attributes which impact on emulation efficiency are presented. Average “instruction emulation rate,” or “emulation power,” is presented as a criterion for the evaluation and selection of host microprocessors for the emulation of target machine CPU’s. The applicability of each of the traditional techniques of performance evaluation and measurement to microprogrammable processors is analyzed in some detail. The technique of synthetic emulation is then developed and presented as a powerful technique for estimating the performance of host-target machine pairs. Simplified models of the target workload, the target machine, the emulation process, and host machine are developed and combined into matrix equations for estimating emulation power. Both single target and multiple target cases are considered and case studies are included. Finally, host-target instruction ratios and host-target bit ratios are briefly considered as indicators of generality and of the relative efficiency with which a given target machine can be emulated by a given host machine.


This paper describes a program package for the design of irredundant single- and multiple-output MOS networks. The program is an implementation of algorithms developed by H. C. Lai for the synthesis of irredundant MOS networks with a minimum number of MOS cells for a given set of completely or incompletely specified functions (synthesized networks having multiple outputs are guaranteed to have a minimum number of MOS cells only under certain assumptions). The output of the program is a graphic representation of the interconnections among FET’s for each MOS cell in the synthesized irredundant network. A listing of the Fortran program is included in the paper.


This report describes the use, structure, and maintenance of an information storage and retrieval system implemented by the Department of Computer Science for its own use in managing information pertaining to both current and past applicants to the department and graduate students presently in the department.


Traditionally, information retrieval systems have provided little more than titles or abstracts of documents in response to user queries. This thesis describes an experimental information retrieval system that provides a framework for research in providing users access to the entire text of documents.

R76-113—Meissner, Loren P., “FOR-WORD, Volume 1” (69 pp., Lawrence Berkeley Laboratory, University of California, Berkeley, California)

The need for communication among those interested in the development of the Fortran programming language was recognized as early as 1974, by such persons as Donald J. Reifer and Guy J. De Balbine. Inspired by these individuals and others, Loren P. Meissner decided in February, 1975, to produce a newsletter for information concerning developments in this language. The first three issues were only one or two pages long, but as more interested persons began contributing material, and as liaison with the ANSI X3J3 Fortran Standards Committee increased, the amount of material considered to be of interest to the readers of the newsletter increased as well. This report is a compilation and reprint of the first six issues of the newsletter, which were distributed between February 1975 and January 1976.

R76-114—Taylor, Richard L., “An Algorithm for Synthesizing and Analyzing Complex Shapes” (13 pp., Institute for Research in Human Abilities, Memorial University, St. John’s, Newfoundland, Canada)

Line drawings of objects are rarely, if ever, completely automated, although the potential advantages from doing so are obvious. This paper presents an algorithm and a set of subroutines in Fortran IV for analyzing and synthesizing the boundary contour of any solid object. We consider that the perspective or orthogonal projection of any solid thing produces a boundary contour which exhibits radial convexity, such that there exists some point from which any ray drawn along a radius at any angle will intersect the contour at either 0 (in the case of a partial contour) or at 1 point. Contours of objects which do not meet this criterion are produced by compound things or collections of more than one thing. It can be shown that these complex contours are composed of two or more components each of which exhibits radial convexity. Accordingly, the algorithm is based on an analogue of the Fourier transform.


This correspondence points out the incorrectness of a theorem in the paper, “On Minimization of Fuzzy Functions,” upon which the proposed minimization technique is based. Two counterexamples are presented. The places in the proof of the theorem which are incorrect are identified. The definitions of implicants and prime implicants of fuzzy functions are also discussed.

R76-116—Tanaka, E. and K. S. Fu, “Error-Correcting Parsers for Formal Languages” (30 pp., Lafayette University, West Lafayette, Indiana)

This paper describes error-correcting parsers for context-free and context-sensitive languages with substitution, insertion, and deletion errors. Furthermore, it is shown that the ability of the proposed parsers can be expressed in terms of the weighted Levenshtein metric.

R76-117—Rhyne, V. T., P. S. Noe, M. H. McKenzie, and U. W. Pooch, “A New Method for the Fast Minimization of Switching Functions” (26 pp., Texas A & M University, College Station, Texas)

The minimization of switching functions involving many variables is a difficult task. This paper presents a new minimization procedure that, by using artificial intelligence concepts, allows this process to be implemented with reduced computational effort. This procedure, designated at the directed-search algorithm, is applicable to both manual and computer-programmed minimization. The details of the algorithm are presented and illustrated by example. Comparative run-times between another minimization program and the directed-search algorithm, as implemented in Fortran, are also given.

R76-118—Pierson, William E. and Paul D. Sigal, “Fault Detection in Combinational Circuits Using A New Approach” (75 pp., University of Missouri, Rolla, Missouri)

Deterministic approaches to fault detection in combinational networks require that a set of inputs be applied to a circuit suspected of being faulty and that the corresponding outputs be compared to those of the non-faulty circuit either by simultaneously applying the inputs to a non-faulty copy of the circuit or by storing the correct outputs of the circuit in some list. This paper proposes that statistical analysis of the outputs be used to detect the presence of faults. Specifically, it is proposed that N inputs be applied to a circuit randomly but according to a set of known and predetermined probabilities. Furthermore, the sample means of the outputs (the number of times the outputs have a value “1” out of N input combinations) are examined to determine if a fault has occurred. It is shown that if the set (or subset) of input probabilities are chosen judiciously, a fault in a combinational network will change the expected value of the means of the outputs significantly.

Order by R-number. Use the Repository order form on page 80.
enough to determine that a fault has occurred. Heuristic criteria for selecting input probabilities, establishing testing procedures, and evaluating test results are discussed. Furthermore, the results obtained when these criteria were applied to specific networks, representative of many circuits currently being used, are given. This paper restricts its discussion to the single, permanent stuck-at-fault assumption.

R76-119—Silverston, Stefan M., “A Stable, One-Multiply-per-Step, Algorithm for Digital Generation of Sinusoids in Real Time” (13 pp., Iowa State University, Ames, Iowa)

A new algorithm for the generation of sinusoids in real time is presented and analyzed. The new algorithm is compared to previously-known algorithms, and shown to be stable and to require only one multiplication per cycle.


This is a first attempt to develop shift-register encryption schemes with non-linear characteristics. We propose some new stream enciphering schemes composed of standard components which are easy to implement and appear to be difficult to break by either a linear attack or statistical analysis.


A high-fidelity video digitizer/processor extracts spatial and gray-scale information from high resolution dynamic video imagery. The 15 MHz bandwidth 60/sec images (fields) are low pass filtered to minimize aliasing and digitized in real-time at 20 m/sec, 9 bits/sample; sequential fields are transferred to memory, retrieved from memory after computer image processing, and reconstituted at up to 40 m/sec into 60 field/sec video of 525, 876, or 1225 lines/frame (two interlaced fields). A/D and D/A converters are synchronized with memory by a high speed interface, composed of a central data interchange (DI), and 8 radially connected I/O ports. The DI connects the data paths of one or more ports, each communicating with one or more external devices. Separate 16-bit, 40 megawords/second, input and output data busses between each port and the DI allow data entry through any port to the DI and output from the DI through any port(s), even the same port from which the input originated. The A/D and D/A are served by a single port. Two other ports communicate with CDC 3500 and CDC 1700 computers. Yet another port controls up to two megawords of core/solid state memory, of which 358K words may be accessed at the full data rate. Storage rates into existing memory of 40 m/sec are achieved for 16-32 m/sec. The DI has dual data busses and throughput processing capability (a binary scaler, high speed RAM, and ALU) which allow simultaneous input of 2 digitized fields from separate sources, gray-scale processing, logarithmic conversion, subtraction of one field from the other, and storage of the difference image, all in real-time. The device is programmed on-line by either computer via 64 microcoded commands.


The design of a high speed digital processor for the sine and cosine functions is discussed. The hardware provides a significant speed advantage over software calculations of these functions. The special processor provides floating point results of 35 bit accuracy within 40 microseconds after the argument is loaded. This is faster by a factor of 5 over the minicomputer software and is comparable to much larger commercial machines. The hardware has the further feature that both the sine and cosine of the argument are calculated simultaneously. The special processor follows the iterative Cordic algorithm in a bit parallel architecture. This configuration uses three fixed point accumulator-shift register units which operate to produce a floating point result. The structure has the advantage that no floating point arithmetic hardware is required. A unique shifting structure is utilized in the arithmetic units to give a high effective shift rate.

R76-123—Claussen, Svend-Erik, Brigitte Madsen, and Erling Madsen, "A Description of the LCODE-Interpreter on RIKKE-1" (38 pp., University of Aarhus, Aarhus, Denmark)

The purpose of this report is mainly to describe how to use the LCODE-system implemented on the RIKKE-1 but also to give a rather detailed description of our experience in developing, testing, and debugging a microprogram of considerable size (we case 1000 microinstructions). One very useful thing we have learned is that the solution to a large and complex problem should be implemented as a structured program in a high-level language, before microcoding takes place. Then, when this program is tested and running, its individual routines may be translated into microcode. Two advantages are gained by this approach. First, the testing of the micro-program will inevitably turn out to be much easier, because performance failures are known not to represent serious logical errors in the major design, but only errors in the translation phase from a high-level language to microcode. Second, the general structure of the program is preserved, whereas any structure in the solution might early vanish in the course of an immediate microcoding process. These advantages clearly override the drawback that the microcode in this way takes up a little more space than if it had been coded directly—without the translation phase mentioned above.

R76-124—Noe, Philip S., “Generation and Selection of Closed Covers for the Tracey Method USST State Assignment Procedure” (12 pp., Texas A&M University, College Station, Texas)

The method for generation and selection of closed covers for USST asynchronous state assignment is presented. The method uses the essential dichotomies of the flow table generated by the Tracey method as a point of origin. The method uses the hash distinction as the assignment column that can be used to assign the y-variables as closed dichotomies and deletes the unnecessary closed dichotomies by comparison with the list of essential dichotomies. Next, the method uses a partial data assignment procedure for selection of the closed dichotomies to generate a closed covering of the essential dichotomies.


In the past several years, there have been several new computer architectures that have come onto the market place which utilize microprogrammed control in interesting ways. Their direction is clear: microprogramming is more accessible than previously and the design of new target systems is being simplified. However, the problem of designing computer systems in today’s market is complicated by the rapidly changing components market. In this paper, our discussion of design principles is related mostly to small computer systems. We first attempt to classify LSIs logic components and to put into proper context the reasons for, plus the advantages of, a proposed outer control concept.

IEEE Computer Society Repository
Cumulative Index 1966-1973

Published by request of Repository users, the cumulative index lists just under 1600 entries for 1966 through 1973. It consists of three parts: an author index, a subject index based on keywords, and a chronological index in the order in which the items have been listed through the years.

Price—$16.00  Member Price—$12.00

To order: Use the multipurpose order form on page 79.