**REVIEWS OF BOOKS AND PAPERS IN THE COMPUTER FIELD**

**B. MICROPROGRAMMING**


The advent of microprogramming is likely to cause a major change in the way we use computers. The reason is that it provides a nearly complete decoupling between the efforts of computer design engineers and programmers. When the engineers provide a microcode that can be changed under program control, they will be able to pass on to the programmer a major portion of the design of the machine instruction set. The question is what the programmers will do with it when they get it.

Helmut Weber's paper describes a particular effort in using the microprogram facility to improve the performance of the IBM System/360 Model 30 for the language EULER. "The essential difference... is that the string language (microcode) reflects the features of the particular higher level language as well as the features of the particular hardware..." The reflection of EULER is taken essentially from the EULER definition and consists of a set of primitive actions that reflect the basic operations of EULER. These actions have a hereditary relationship to the English Electric KDF9 and Burroughs B5500 as well as to the accumulated lore of ALGOL-60 implementations.

Did the programmer succeed? In the opinion of the reviewer, yes, and spectacularly so. Unfortunately, it is very hard to get believable comparisons over sets of programs that the readers will agree are representative. FORTRAN advocates will insist on solving linear equations and EULER advocates will insist on recursive factorials. The author offers only an opinion that his microcode is an order of magnitude more efficient, based on a comparison for the operation of logical AND, and later he says his compiler is card-reader limited. Neither is very satisfactory and the reader is left wondering when someone is going to give a sufficiently convincing demonstration of this "order of magnitude" to change the course of a manufacturer.

What is left unsaid in this paper is that there is another "order of magnitude" to be gained, that being in the ease of programming a machine with a really good and perhaps even a changeable language. It is hard to see how we can claim the title "computer science" when we don't even have control over the notation in which we state our problems. The deferral of the design of the instruction set right down to the workaday programmer may yet be the most significant development. The author states "A proliferation of interpretive languages and the development of microprogrammed interpreters can be justified when better tools are developed to reduce the cost of microprogramming." We take this as a warning that microprogramming, like machine language programming before it, is difficult. And we pass the warning on to the computer designers that their critics of tomorrow will be the microprogrammers. Even a microcode has to be usable. In this regard, it would have been helpful for the author to record how long it took to complete his microprogramming task.

The reader will not find enough detail duplicated in the author's work, but this is not a serious drawback. Such a catalog would have elucidated mostly the idiosyncrasies of the EULER language and the Model 30 microcode. It is, of course, important to search for "ideal directly interpretable languages which correspond to higher level languages," but the reviewer recommends the parent paper cited above for this purpose. A less precise discussion of these ideas has been given by W. M. McKeeman.¹

W. M. McKeeman
Stanford University
Stanford, Calif.

**C. ANALOG AND HYBRID COMPUTATION**


A simple model of a hydroelectric generation system is described in this paper, as well as its implementation on an analog computer. Dynamic optimization of the system, its implementation using analog and digital computers, and applications of the self-optimizing system are briefly discussed.

Larger power pools, more complex interconnections, and conflicting needs for water imply the need for more efficient management of hydro power systems. The simulation approach allows analysis of the interrelations among energy availability, loads, costs, weather, political constraints, etc. The analog model discussed consists of three reservoirs, streams feeding and interconnecting them, three generators, four electrical loads with daily or long-term variation, and transmission lines with losses. A performance index was defined as the total energy loss. The optimization strategy consisted of adjusting generator demand to minimize the total energy loss.

Ten runs were made, recording water levels, power generated, energy losses, and power demand. The first six runs predicted over an interval of one week and included the nominal open loop case, a manual scan in generator demand 1 and 2, the optimizing loop closed around generator demand 1, the same with manual scan of generator demand 2, and the reverse. These results illustrated unilateral coupling (not surprising) in that a scan of demand 2 had little or no effect on demand 1, but a scan of demand 1 did affect demand 2. The remaining runs included one-month, one-year, and ten-year prediction intervals, with the loop closed on generator demand 1, an exponential load on generator 3, etc. No surprising results were obtained, but optimizing settling time did increase considerably.

Initial conditions for the model were water level in the reservoirs and the reaches, and snow level. Parameters included the heat for melting the snow, rain, generator loads, spillway levels, and optimizing circuit gains. Factors that could be added to provide a more complete model include capital cost, interest rates, recreation needs, agricultural and commercial needs, and statistical inputs such as weather, population demands, industrial use, etc. Applications include a digital computer mechanization of a much more complete model giving hourly optimal operating information, hourly reports on energy availability, computing trade-offs in location of new industry, and computing cost of proposed changes in operating philosophy.

Altogether, this paper is a well-written description of a demonstration program, illustrating the concept of automatic plant optimization. The following comments reflect areas where the paper could be more useful to the reader.

1) A mathematical model as such was not presented. While most of the model could be deduced from the analog diagram, there is lack of rigor in tasks such as the computation of reach water level from flows. A precise mathematical model would make such a report much more useful, especially to the non-computer oriented.

2) No attempt was made to validate the computer model. While this was probably beyond the scope of the project, a discussion of this vital aspect would have been in order.

3) No discussion was made of the impact this demonstration has made on management, and what future plans have been made for its development and exploitation.

DON DARMS
Electronic Associates, Inc.
Rockville, Md.


This paper describes work carried out by the author while working for his Ph.D. degree. The work was done at the University
