register. Functionally the machine is completely described by a list of all possible transfers, with the timing and control signals on which they occur. "The design process is then divided into three phases: (1) the system design, which sketches in the general configuration of the machine and specifies the general class of hardware to be used; (2) the structural design, which describes the system in terms of transfer relations; and (3) the logic design, which realizes the transfer relations by means of Boolean equations." This design method is developed largely by examples in five chapters, which include descriptions of a simple general-purpose computer, a radar data digitizer, and a digital differential analyzer.

Five chapters comprise an introduction to switching theory. One introduces Boolean algebra informally by the method of perfect induction, while another develops it axiomatically via Boolean rings and fields. One on minimization techniques describes the Quine-McCluskey method and extends it to multiple output networks. Sequential machines of the Moore and Meahly types are described by flow tables and state diagrams. In the last chapter, flow tables are reduced by merging equivalent or compatible states and a final section is devoted to Turing machines.

At the end of each chapter is a set of problems and a usually extensive bibliography that will lead the reader on to topics not covered in the book.

This very readable book presents probably as uniform and systematic a technique as has been devised for the design of digital machines; though, of course, a host of system design considerations lie outside the scope of the method.

R. D. Elbourn

National Bureau of Standards
Washington, D. C.

93[Z].—Bernard A. Galler, *The Language of Computers*, McGraw-Hill Book Company, Inc., New York, 1962, x + 244 p., 23 cm., Price $8.95.

The reviewer must confess to a certain excitement as he first opened this book. What could be *the* language of computers? As it turns out, the choice of the title can only be laid to literary license, perhaps with the marketplace in view. How else could one account for the page headings opposite one another on pages 196–197: "The Language of Computers"—"Other Computer Languages"? So let us be honest at the outset and recognize that this is a book written to promote a particular computer language, namely, MAD.

The author has written "for the person who is interested in learning how problems are solved on electronic computers." To achieve his purpose, he has organized the body of the book into cycles consisting of the formulation of a problem, the development of an algorithm for solving the problem, discussion of the special linguistic requirements imposed by the algorithm on the programming language that is being used (essentially MAD), and finally, as the culminating phase, the display of a program in MAD that effects the desired solution. By this procedure the introduction of the various features of the language takes place in a natural and persuasive manner, and the student gains practice in the mode of thinking that one must adopt in order best to exploit the contribution of the computer.

Somewhat different critical standards must be used to measure those portions of the book that are substantially independent of any particular programming language and those that are MAD-oriented. To those in the first category, comprising, say, 60% of the first ten chapters, the author has brought the full power of a talent for vivid, thorough exposition. Here we find rapport with the reader established at once, along with the clear impression that the writer is sympathetically aware of the difficulties that face the novice in the field. As by-products we have little introductory essays on certain topics, as, for example, arithmetic congruence, random numbers, sorting, switching functions, and the solution of systems of equations—gems in their own right and skillfully set into the main structure.

However, the author's inclination toward hucksterism is manifested not only in naming the book, as already noted, but in titling his chapters as well. One feels that the material under "The Secret-code Problem" was contrived to fit the title rather than included because of its intrinsic suitability, and to call his excellent ninth chapter "A Program to Produce Programs" instead of "A Program to Produce Network Descriptions" smacks of hypocrisy, which the opening remarks of that chapter fail to mitigate.

One's estimate of the success of the portions of the book in which MAD is described or used will naturally be colored by his own bias as to the merits of MAD vis-à-vis alternative languages. Under "Other Computer Languages" we find FORTRAN and ALGOL briefly treated, and there are appendices giving a feature-by-feature translation from MAD to each of these insofar as translation is possible.

We are entering an era of books about programming in specific programming languages, and those who are building libraries of such books will want to have MAD represented by this interesting, well executed work.

R. N. Goss

Navy Electronics Laboratory
San Diego 52, California

94[Z].—G. M. Hartley, *An Introduction to Electronic Analog Computers*, John Wiley & Sons, New York, 1963, vii + 155 p., 19.5 cm. Price $4.50.

The analog computer is presented in its simplest form, as a tool to solve linear differential equations with constant coefficients. As a result, the operational amplifier is the only active device on which the book concentrates. The fact that the author is an engineer is apparent because the bulk of the material analyzes the characteristics of d.c. amplifiers, while comparatively little material is spent on the application of equipment.

The level of the book is such that it could easily be understood by a junior in electrical engineering. With prerequisites of differential equations and a course in electronics, the material in the book could be covered in eight one-hour lectures.

The first two chapters are background in nature; the first deals with the history of the analog and digital computers, while the second is a general survey of applications for various types of computers. The essence of the book is contained in the next four chapters. Chapter 3 illustrates the role of the operational amplifier in performing mathematical operations. Chapter 4 describes in detail the programming, scaling, and wiring of a second-order differential equation with constant coefficients for a typical computer. This chapter also touches on time scaling, ampli-