Book Reviews

The following reviews were selected from those recently published in various IEEE TRANSACTIONS *and Group/Society Magazines and Newsletters. They are reprinted here to make them conveniently available to the many readers who otherwise might not have ready access to them. Each review is followed by an identification of its original source.*

Operational Amplifiers and Linear Integrated Circuits—Robert F. Coughlin and Frederick F. Driscoll (Englewood Cliffs, NJ: Prentice Hall, 1982, 376 pp.). *Reviewed by Phillip E. Allen, Department of Electrical Engineering, Texas A & M University, College Station, TX 77843.*

In their preface, the authors state that their purpose is "to show just how easy they (op-amps) are to use" and to let readers "gain confidence in using op-amps before pushing performance to its limits." The authors indicate that "the material is suitable both for nonelectronics specialists who just want to learn something about linear IC's and for electronics majors who wish to use linear IC's."

The authors have created a book which flows smoothly, and carefully avoids problem areas. The authors develop an increasing level of capability and complexity, building concepts on previous ones. A particularly helpful aspect is the extensive indication of current flow in the op-amp circuits. This helps the reader to keep track of where signals actually flow.

As a first-time introduction to the study of op-amps, the book could create problems, mostly because the reader may become overconfident of his ability to use the op-amp properly and run into unanticipated problems. The reader learning about op-amp circuits for the first time who runs into a problem using op-amps, will most likely be unable to solve the problem with the material contained in the book. Formulas are given without derivation and are not general. If a particular circuit does not fit an application, the reader may be unable to modify it to meet his needs.

All the circuits used have been built and tested, but I found it surprising that more photographs or actual waveforms were not included to give the reader even a better feeling of what to expect. Some of the waveforms drawn by artists are misleading (for example, slew rate). The material presented is representative of op-amp circuits and includes other linear integrated circuits such as analog multipliers, timers, and voltage references. The authors should include a section on the use of switches in op-amp circuits. I would have also expected the illustration of the use of an op-amp from a single supply without the use of blocking capacitors. All concepts are carefully defined, though in the somewhat simplistic terms which would be expected.

The authors succeed in meeting their claims in the preface. The student should "gain confidence in using op-amps before pushing performance to its limits" but I am concerned with what will happen when the limits are reached. The book is suitable as a text in a technology course if the student is aware of the problem of overconfidence. It would only be useful in an electrical engineering curriculum as a reference. The book is very well suited as a source of application ideal for those with a solid op-amp background.

Reprinted from *IEEE Circuits and Systems Magazine*, vol. 5, no.1, pp. 20–21, March 1983.

Turtle Geometry—Harold Abelson and Andrea A. DiSessa (Cambridge, MA: MIT Press, 1981, 477 pp., $22.50). *Reviewed by Dave Bones.*

Designed as a textbook for high school and undergraduate college students, *Turtle Geometry* demonstrates a new way to "experience" math through the use of inexpensive personal computers. This innovative text demonstrates how computers in the hands of students can be used as powerful tools for self-directed learning and understanding. This is the central and recurring theme of the book, which is in fact subtitled "The Computer as a Medium for Exploring Mathematics." The authors are both MIT faculty members in the school's Study and Research in Education Division, where this material was developed.

The turtle in this book is a small graphical creature, much like a cursor, who traces paths according to programmed sequences of simple FORWARD steps and LEFT/RIGHT turns. From the turtle's point of view, any path can be represented by the turtle program required to generate it. In fact, the turtle has no knowledge of coordinate systems, axes, or other such external frames of reference.

This procedural viewpoint provides interesting insights into the topology of curves and surfaces and, of course, lends itself well to programming experiments. The long equations and symbolic manipulations found in conventional math texts are replaced in this book by short programs and process-oriented formulations.

Although much of this material can be studied with only pencil and paper, access to a programmable computer with graphics output is necessary to try the many suggested programming experiments and projects. An inexpensive personal computer with a graphic CRT display is ideally suited to running these experiments. (The authors assume that the reader has had some exposure to fundamental programming and Cartesian geometry.)

The first chapter opens with an explanation of basic Turtle Procedure Notation, a simple procedural language used throughout the book. This is followed by a brief examination of the differences between turtle geometry and more conventional coordinate geometry. Very simple programs for tracing polygons are developed and then used in discussions of closure, looping programs, and symmetry.

The reader and turtle graphics explore a number of different phenomena and situations in the second chapter. Ordinary animal behavior is modeled with "smell" and "sight" feedback, and spirals and branching trees are studied as examples of biological growth. The authors also look at recursive designs such as snowflakes and Hilbert curves. This chapter skips around and bears little relationship to the rest of the book, but it does serve to stimulate mathematical experimentation.

In the next three chapters, various mathematical tools are introduced for use during explorations in subsequent sections of the book. The third chapter introduces vector analysis in a fairly traditional manner, including coordinate representations of vectors and methods for displaying vectors. Vector addition, multiplication, and rotation are presented, as well as parallel and perspective projections of three-dimensional vectors. In addition, the discussion of vectors serves as a vehicle to illustrate linearity and state.

The topology of planar curves is explored in the fourth chapter, with discussions of topological types and deformation, and an algorithm for escaping from a maze is developed using the topological tools. Then in the fifth chapter, the turtle is introduced to curved surfaces, including spheres, cylinders, cones, and tori. Turning and curvature are discussed at length, serving as cornerstones for much of the material that follows.

In Chapters 6 through 8, the turtle is at its best. It explores the surfaces of cubes in Chapter 6, revisits spheres in Chapter 7, and moves on to piecewise flat surfaces in Chapter 8. The latter also includes several special surfaces such as Möbius strips and Klein bottles. Each of these chapters begins with ways of representing the respective surfaces in a computer in a manner suitable for experimental turtle wanderings. The student is then strongly encouraged to experiment by trying various turtle programs to observe and understand the resulting paths and the implications upon the surfaces. These chapters demonstrate, more than any of the others, just how computers can be used as learning tools.

Making a fitting finale, the ninth chapter takes the earlier discoveries about curved surfaces and extends them to curved space and time for use in exploring Einstein's general theory of relativity! Appendix A contains a cursory description of Turtle Procedure Notation and Appendix B explains how to translate turtle programs into other languages, including Basic and Pascal as implemented on Apple II personal computers.

Throughout the book there are numerous exercises carefully designed to provoke thought and suggest programming projects. Additionally, a section at the end of the text provides hints and answers for many of the problems presented by the authors.

Turtle Geometry is not a substitute for conventional geometry; rather it is an extension into subjects usually left for more advanced studies. It demonstrates the effectiveness of inexpensive computers as learning tools and as such should be of particular interest to educators and students. In addition, it should be useful to anyone interested in this procedural approach to the topology of curves and surfaces.

Reprinted from *IEEE Computer Graphics and Applications*, vol. 3, no. 1, p. 62, January–February 1983.

Algorithms for Graphics and Image Processing—Theo Pavlidis (Rockville, MD: Computer Science Press, 1982, 400 pp., $26.96.) *Reviewed by R. M. McElhaney, Graftek.*

Titles of books and their contents make curious bedfellows at times. Glancing at the cover of this text, I expected to find a very technical treatment of selected subjects in the graphics and image processing fields—subjects enhanced by a generous sprinkling of coded algorithms that practitioners in the field would find useful. What I actually found, however, both surprised and disappointed me.

Algorithms for Graphics and Image Processing is filled with a number of useful techniques, yet its long-term fate is to sit dusty and neglected on a reader's shelf. This is only an apparent contradiction, and it vanishes when one realizes that this is not one book but actually two: one about classical image processing and one whose subject is traditional interactive graphics. To my mind, it is indeed unfortunate that this text could not have appeared as two separate volumes. I would have kept the text on image processing, giving it a special place in my own reference library, and given the other volume back to the publisher.

The author makes no mention of the historical fact that image processing and interactive graphics represent two separate evolutionary graphics developments. Interactive graphics began with calligraphic (line-drawing) CRT systems based upon refresh-vector hardware and later the DVST (storage tube). Admittedly, this field has recently begun to make use of raster technology, but primarily still uses the technology in line-drawing applications.

Image processing, however, instead of dealing with lines and points and continuous display devices, was, and is, based upon a rectangular array of picture elements. Thus, imaging *requires* discrete output devices such as a frame buffer or pixel memory, and all of the algorithms presented in the first half of the book deal with the discrete nature of the output device and with the pixel as the basic unit of picture information.

The transition in the text between these two subjects is sudden and artificial. Chapter 9 is a well-written treatment on "thinning algorithms"—a popular subject in pattern recognition—while Chapter 10 launches into a discussion of interpolating polynomials in curve fitting. To his credit, the author's introduction to curve fitting *is* an attempt to blend these two subjects; he discusses the display of curves on dense pixel grids, which, unfortunately, has nothing whatsoever to do with the subject, even as he presents it. From this point on, pixels are never again mentioned significantly, whereas they had been the primary subject of discussion in the book's first half. In fact, the author never mentions graphics display devices again, except for a few references to curve-generating circuits.

The image processing half of this text requires, in the author's words, "a first course in signal processing" (with its heavy emphasis on the Fourier transform) for a complete understanding of the book's first five chapters. This is fine and is to be expected. But signal processing is a background generally held by engineers, physicists, and applied mathematicians; it is not one commonly found in those who study, or work with, the traditional applications of interactive graphics.

My point is not that the book is too heavy on background requirements, but that it attempts, quite unsuccessfully, to present two disparate subjects within the context of "graphics." In fact, all that a reader such as myself can see in the way of common ground between these two subjects, especially from the author's presentation, is that they both require graphics display devices.

Actually, these two disciplines are beginning to draw together, due principally to the recent availability of powerful, high-resolution raster devices and the resulting increase in the number of applications within traditional interactive graphics that require imaging-quality devices.

For example, solids modeling, a popular new CAD/CAM technique, requires the use of high-quality, color raster devices and, indeed, picture processing techniques to display shaded 3-D models with shadows and realistic surface textures. These developments are very exciting and I personally feel that the author missed a unique opportunity to present these two subjects in a unified manner.

A second fundamental criticism I have is that, although the imaging sections of the book are really quite well-written and would tend to be useful to a practitioner in the field, the sections on interactive graphics would not. The author attempts to treat a rich and complex subject with mere technical highlights. In my opinion, this treatment neither interests the serious reader nor provides any truly useful content for the professional. The algorithms presented by the author are in a form called "structured English." and the step from his programs to those that can actually be used is a very large step indeed.

One of my more useful reference books (on numerical methods) contains entire Fortran programs illustrating some of the more complex algorithms, and I have successfully keyed these into my computer directly from the book and had them working in a short time. What I needed, obviously, was not an education but an algorithm. In fairness, one could not expect the author to have coded many such algorithms, since most of the simple and elegant algorithms are properly left to the reader's ingenuity. A more explicit presentation, however, would save the practical reader a great deal of time.

In the interactive graphics section of the book, the author presents all of the "right" topics—curve fitting and display, parametric curve and surface definition, polygon clipping, and 2-D and 3-D display transformations, etc.—but he does not delve far enough into any of these to be truly useful.

There are a number of interesting illustrations in the imaging section of the book, with many photographs nicely illustrating the imaging algorithms. The "graphics book," on the other hand, has a noticeable lack of illustrations and contains no photographs at all. In Chapter 13, titled "Surface Fitting and Surface Display," there is not a single illustration of a surface! With all of the resources available to the author as a long-time practitioner in graphics, there is really no excuse for this omission.

From reading these "two books," I have personally gained the clear and unmistakable impression that, while the author clearly lives and breathes the subject of image processing, he is merely "writing about" interactive graphics; thus, my recommendations of this book are obviously mixed. I heartily recommend it insofar as it is a book on image processing. It is well written, well ilustrated, easy to read and understand, and offers the reader insights into the techniques and applications that make up this fascinating field. It should be in everyone's reference library.

As a book on the techniques of interactive graphics, however, it is a virtual failure, and I cannot recommend it. It does not approach the standards the author himself sets in the first half of the book, and readers can easily find a number of other books on this subject which are far, far better than this one.

Repreinted from *IEEE Computer Graphics and Applications*, vol. 2, no. 6, November-December 1982.

Introduction to Computer Organization—Ivan Tomek (Rockville, MD: Computer Science Press, 1981, 456 pp., $21.95). *Reviewed by William J. Tracz, IBM Federal Systems Division.*

Too often, computer science students have a limited understanding of computer hardware, and even programming professionals have little relevant hardware background. Recognizing that the needs of these two groups differ from those of engineers, Ivan Tomek has written a timely and significant contribution to the academic arena. *Introduction to Computer Organization* covers all the topics recommended in CS4, the 1978 ACM curriculum proposal bearing the same title as this self-contained textbook. The author has also provided a workbook containing solved problems and examples to augment the main text, a combination that appears ideally suited for a one- or two-semester course on the principles of computer operation. Furthermore, the material is presented in such a cohesive, conversational manner that even a layman could read it profitably.

Roughly half (three chapters) of the book deals with computer architecture. The concepts are introduced, then supported by a wealth of examples, including the Motorola 6800 embedded in the Heathkit ET-3400 microcomputer trainer, the DEC PDP-11, the Univac 11/60 system, and the CDC 170 series. This section also discusses current technology trends at length, overviewing VLSI, Josephson junctions, and bubble memories to name a few.

The presentation of combinational and sequential circuits, which makes up about one-third (three chapters) of the book's total content, is supported by realistic examples using the 7400 logic family of TTL chips; these real-life examples subtly and very effectively present the digital designer's perspective. For the third major focus, two chapters are devoted to number representations and coding. Additional topics presented that are not part of CS4 are reliable design, fault analysis, and testing. Also of significance is a chapter on hardware simulation that introduces circuit design and analysis from a programming perspective, thus relating hardware design in software terms and exposing the student to rudimentary design-automation concepts. Finally, the textual material is followed by a detailed glossary.

This book meets the author's objective of presenting a "realistic, up-to-date picture of the current technological state." In combination with the workbook, it provides a truly significant resource that computer science students or programming professionals can draw upon to gain insight into computer hardware and architecture.

Reprinted from *IEEE Computer Magazine*, vol. 16, no. 2, p. 125, February 1983.

T_EX and METAFONT, New Directions in Typesetting—Donald E. Knuth (Providence, RI: American Mathematical Society, and Bedford, MA: Digital Press, 1979, 362 pp., $12.00). *Reviewed by Richard Furuta, Department of Computer Science, University of Washington, Seattle, WA 98105.*

This book is probably already in the library of almost every computer scientist who is interested in the design of document formatting and typesetting tools. It describes two of the most interesting and influential of these modern systems: T_EX, a system for formatting and typesetting