COMPUTER APPLICATIONS IN ELECTRICAL ENGINEERING SERIES

Franklin F. Kuo,
Editor

RAYMOND M. KLINE

Digital Computer Design

DIGITAL COMPUTER DESIGN

RAYMOND M. KLINE

Washington University
St. Louis, Missouri

Prentice-Hall, Inc., Englewood Cliffs, New Jersey

Library of Congress Cataloging in Publication Data

KLINE, RAYMOND M.
Digital computer design.

Includes bibliographies and index.
1. Electronic digital computers—Design and construction. I. Title.
TK7888.3.K56 621.3819'58'2 76-9865
ISBN 0-13-214205-8

10 9 8 7 6 5 4 3 2 1

Printed in the United States of America

PRENTICE-HALL INTERNATIONAL, INC., *London*
PRENTICE-HALL OF AUSTRALIA PTY, LTD., *Sydney*
PRENTICE-HALL OF CANADA, LTD., *Toronto*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*
PRENTICE-HALL OF SOUTHEAST ASIA PTE. LTD., *Singapore*

PREFACE

This text presents a broad, though substantive, introduction to the three major levels of computer hardware design: architecture, logical design, and digital electronics. In addition it develops sufficient software concepts to establish the necessary background for making hardware design decisions. A major goal of the text is to provide a modern, quantitative overview of hardware design for people in all computer areas. Among the new topics presented are: an introduction to microprocessors and control memories, development and application of a register transfer language, use of heuristic-design techniques, investigation of various processor structures through simulation models based on FORTRAN, and application of MOS and bipolar integrated circuits. The text stresses general design tools and hardware principles rather than current specialized devices which may soon be discarded. (The treatment of NAND/NOR circuit design methods in Chapter 4 is an example of this approach.) On the other hand, the text includes practical details and well-recognized contemporary designs.

I assume that the *average readers* will be second- or third-year college students of electrical engineering or computer science. Many will have completed a basic course in a procedure-oriented language, like FORTRAN or PL/I, and a one-semester course in electrical networks. However, such courses are not *strict requirements*, for I have indicated those specialized sections which the reader may omit or cover less thoroughly without jeopardizing his understanding of later chapters. Moreover, I am assuming no advanced mathematical background, and I have presented sufficient details so that the book is suitable for individual study. The text contains the depth required by those readers majoring in computer technology, yet it is flexible enough for those who are in other areas of engineering or science and require only a limited introduction to digital systems. The text includes sufficient material for a three-hour course of one semester or two quarters in length; but it may be employed for a two-semester course with a limited amount of supplementing from the bibliography. In addition, it is an appropriate book for introductory computer courses such as those recommended by the COSINE Committee (Commission on Education of the National Academy of Engineering) and by the Association for Computing Machinery.

Undoubtedly, many students will later take specialized courses in the various topics covered by this text. However, I believe that a broad introduction to digital computers, with depth in some areas, gives the student the perspective and background required to assimilate more advanced courses. My text, therefore, is intended to foster the maturity and experience these students require to make the advanced course meaningful. As a specific example, most of our computer engineering students at Washington University follow their introduction to computer hardware by a three-hour course devoted to switching theory. Chapters 4 and 5 on logical design, while not intended to be exhaustive treatments, are complete enough that our switching course is able to progress at a more rapid pace and on a more advanced level.

I have observed that introductory computer courses often do not present the same challenge to college students that many of their other courses do; thus, students frequently do not learn as much and are not as highly motivated. Certainly, there is no need to create artificial challenges, but it is one of my objectives to present the material in sufficient depth to be useful in practical design situations as well as to command the interest of the most promising students. For several years I have taught undergraduate digital computer courses following this objective, and my text has been compiled from that experience. Preliminary versions have been extensively tested in a three-hour introductory course on computer organization and logical design.

Students and colleagues, too numerous to list, have been my teachers during the preparation of this book; their questions, suggestions, and ideas were indispensable. I am indebted to Russell Pfeiffer, Chairman of our Department of Electrical Engineering until his tragic death, for the encouragement he provided me during the writing process. Debbie Hawley did a praiseworthy job of typing the very early portions of the manuscript. Debbie Fivecoat did all the other typing and drafted the figures, and she deserves a special note of appreciation for accurately translating my very rough pencil notes and patiently completing many manuscript revisions. Finally I wish to acknowledge the unique climate for computer research, development, and teaching present in the Department of Electrical Engineering, the Computer Research Laboratory, and the Biomedical Computer Laboratory at Washington University for strongly influencing my thinking.

R. M. K.

CONTENTS

1 AN OVERVIEW

It is always very helpful to gain perspective about a new subject before design techniques and mathematical details are considered. Thus, this chapter emphasizes the role of computers in our society, their practical importance, their evolution, and other nontechnical aspects, while in all the following chapters, the emphasis is on engineering analysis and design. In addition to the traditional computer hardware topics, several innovative features will be found in later chapters. Typical members of this group are the heuristic design methods presented in Chapter 5, the register transfer logic developed in Chapter 7, and the microprocessors treated in Chapter 10.

Digital computers have assumed a prominent place in our society, and their value in business and industry is without question.[6,7,11] However, many people still have a distorted view of them. Some picture computers exclusively as big adding machines, when in reality they currently perform extremely complex logic; general nonnumerical tasks are also becoming increasingly important.

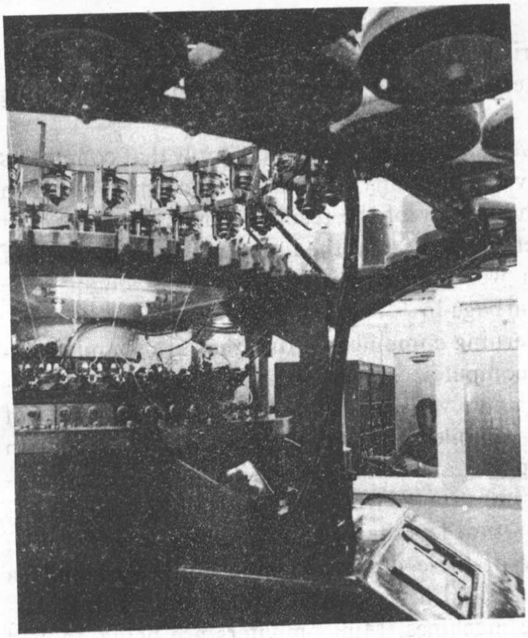
Typical of this trend is the application of computers in medical research and treatment. An interesting case in point is the development of "artificial hearing" for tens of thousands of deaf people who cannot be helped by conventional hearing aids. In an experimental program, scientists from the Ear Research Institute in Los Angeles and the University of Utah in Salt Lake City reported that a deaf subject was able to sense "sounds" produced by a PDP-8/F minicomputer. Figure 1.1 shows a 62-year-old engineer named Joe, deaf since infancy, who is participating in experiments to develop an artificial ear. Touching a key or combination of keys on the keypad in his lap, Joe causes the PDP-8/F, left side of photo, to send signals at selected frequencies to electrodes implanted in his inner ear. The computer provides a flexible system for conducting a wide variety of tests, eliminating the need for building special equipment for each experiment. When the experiments are completed, a portable artificial ear will likely be constructed using a subminiature computer, a *microprocessor*, not much larger than a conventional hearing aid.

Another example of an important nonnumerical application is the computerized knitting machine shown in Figure 1.2(a). One of the problems in the textile industry is the long time required for development of new fabric

Figure 1.1 Patient and Computer-Controlled Artificial Hearing System (Courtesy of Digital Equipment Corporation).

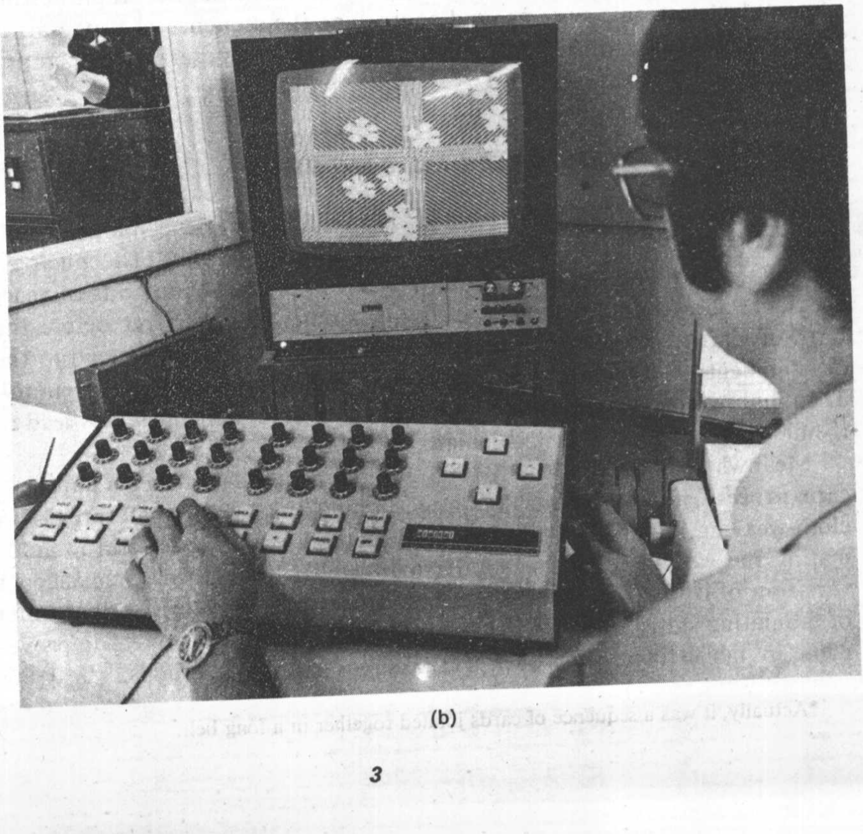
patterns on conventional knitting machines. By using a PDP-11/20 minicomputer [background at the right of Figure 1.2(a)] to control one of its knitting machines (left foreground), the Uxbridge Knitting Mills of Massachusetts have been able to reduce the fabric development time to a matter of hours, or even minutes for less complex tasks. The computer-aided pattern design begins with sketches made, at the operator's console, on an electronic tablet coupled to a color television monitor by the computer. [See Figure 1.2(b).] As the work progresses, the computer transforms the information displayed on the monitor into stitch data recorded in the core memory of the PDP-11/20. When the designer wants to examine a piece of the new fabric, a brief set of instructions is given through the console, and the stitch data in the memory provides control of the knitting machine to produce a sample. Should corrections be required, the designer can recall the pattern to the television monitor and indicate the adjustments with the electronic tablet. When the designer is satisfied with the modified pattern, he allows the computer to resume production. A much more complicated procedure is required with older methods.

Other representative nonnumerical tasks are automated typesetting,



(a)

Figure 1.2 A Computer-Aided Fabric Pattern Design System: (a) Knitting Machine Controlled by a PDP-11/20; (b) Operator's Console and Television Display (Courtesy of Digital Equipment Corporation).



(b)

computer-aided drafting, derivation of theorems, and weather prediction. The reader should consult References 2, 5, 6, and 7 for details concerning these applications.

Popular magazine and newspaper articles frequently imply that computers are some kind of magical device. In reality, they are based on well-known engineering principles, and the quality of the results is critically dependent on the quality of the computer's logic design, the input data, and the skill of the programmer. (There is a well-known motto in computer engineering which aptly expresses this: Garbage in . . . garbage out!)

Another false notion concerning computers is that they are found exclusively in multimillion-dollar computer centers; in reality, a small general-purpose machine, a *minicomputer*, with a teletypewriter, 12,000 words of memory, and a FORTRAN compiler is currently available for less than $5,000, and surprisingly the prices are coming down. So-called microprocessors, and pocket calculators constructed from large-scale integrated circuits and costing two orders of magnitude less than a minicomputer, are available; but limitations on the ability of these devices to independently handle a complete spectrum of computational tasks prevents them from being classified as truly general-purpose machines. Minicomputers are being applied to such tasks as automatic monitoring of industrial processes, rapid testing of complex electronic systems, and controlling the sequence of programs at television stations. (In 1975, more than 30,000 minicomputers were operating in the United States.)

1.1 THE EVOLUTION OF DIGITAL COMPUTERS[1]

Although the digital computer itself is a quite recent development, it has had a complex evolution that goes far back into the history of technology. One of the earliest sources of digital concepts occurred thousands of years ago in Hindu-Arabic arithmetic notation. Somewhat later, yet before the birth of Christ, the odometer was invented, and this eventually lead to the development of the mechanical adder and multiplier in the 17th century. Then, improvements in machine tools and other developments finally lead to the first mass-produced desk calculator in 1911.

Meanwhile, in approximately 1800, Joseph Jacquard perfected a punched-card process* for automatically programming the location of threads in cloth-weaving looms. The punched-card method was so successful in automating the weaving process that Herman Hollerith decided to consider a variation of it to solve the massive data-processing problem (for that time) of tabulating statistics for the 1890 census. The result was a great improvement in tabulating speed over the previous manual methods. (In 1896,

*Actually, it was a sequence of cards jointed together in a long belt.

Hollerith formed his own company to manufacture punched-card equipment, and this later became part of IBM.)

Of all the contributions to the computer, by far the most remarkable was the contribution of Charles Babbage[4,8] (1791–1871), who was a professor of mathematics at Cambridge University. He conceived many of the basic ideas of the modern digital computer over 100 years before they were perfected. His *difference engine*, shown in Figure 1.3, was intended to prepare mathematical tables of various types by accumulating differences through mechanical means. The apparatus shown in the picture is capable of producing tables to 20-place accuracy. In the process of his work, however, Babbage conceived of a much more general machine, the *analytical engine*, having several of the properties of a modern computer. Although his theory was sound, the mechanical technology of his day was not capable of supporting his very ambitious ideas, and his analytical engine was never completed.

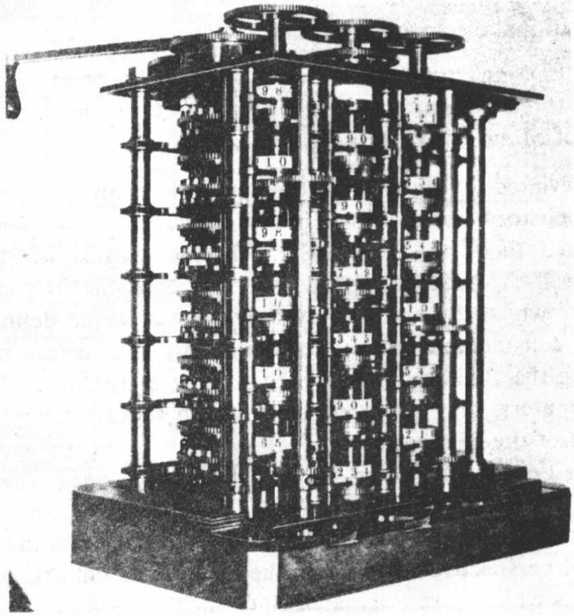


Figure 1.3 Babbage's Difference Engine (Photo Courtesy of IBM).

Although there were improvements in desk calculators, in punch card tabulating machines, and in other calculating devices between 1900 and 1937, no real breakthroughs occurred. Then, from 1937 to 1950, revolutionary changes occurred in such rapid succession, by so many different people, and in so many different places that it is impossible, in this short chapter, to give credit to all those concerned. Some of the major contributors were:

(a) John Atanasoff and a colleague, Clifford Berry, at Iowa State College during the period 1937–1942 demonstrated an experimental special-purpose electronic digital computer for solving simultaneous linear equations.

(b) During the early 1940s George Stibitz, at Bell Telephone Laboratories, built several special-purpose digital computers based on relay technology. He was one of the first to implement floating-point arithmetic.

(c) The Mark I relay computer was completed in 1944 by Howard Aiken at Harvard. It is considered to be the first general-purpose machine.

(d) The ENIAC, Electronic Numerical Integrator And Calculator, was completed in 1946 by J. Presper Eckert and John W. Mauchly of the Moore School of Electrical Engineering of the University of Pennsylvania. Although somewhat similar in structure to the Mark I, it employed vacuum tubes instead of relays; thus, it became the first operational electronic computer.

(e) The EDSAC, Electronic Delay Storage Automatic Computer, the first stored-program computer, began operating in 1949 at the University of Manchester in England.

An abbreviated family tree for the computer is shown in Figure 1.4. It includes the history just described, some important events since 1950, and certain related facts not previously mentioned. Most of the individual entries are self-explanatory; a few, such as core memory and assembly language, for which the reader may only have a vague definition, will be explained in considerable detail later. More historical details may be found by consulting the Bibliography.[3,4,8,10] (Because of the multiplicity of commercial computers, only a few of those which represent landmarks in the development of the field appear.)

From the density of dots in Figure 1.4, it may appear that the activity peak in the development of computers was reached around 1960. This is misleading because a vantage point in the 1970s does not allow one the proper historical perspective to evaluate the long-range importance of most of the techniques which were originated since the early 1960s; hence, only such obvious developments as Large-Scale Integration, LSI, have been added to the latter portion of the chart. An example of a development which may be later inserted in the chart is the highly parallel machines such as Illiac IV. (It was proposed in 1968 and completed in 1972.)

A number of items, such as the slide rule and Babbage's difference engine, are not directly connected into the mainstream. Although they had a role in the development of computer science, their contribution to the digital computer can not be identified through a specific single path.

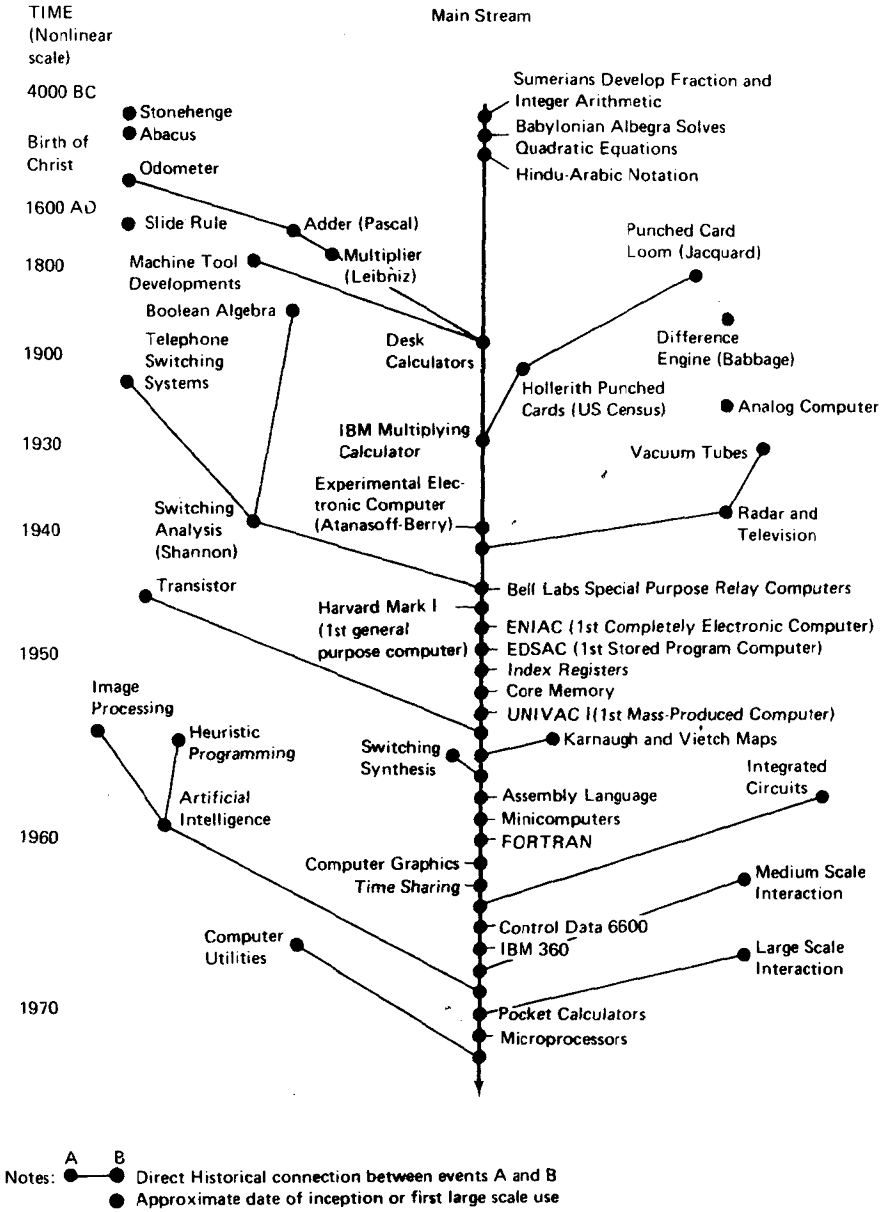


Figure 1.4 A Brief Chronology for Digital Computers.

1.2 THE GROWTH OF COMPUTER TECHNOLOGY

Having just considered the general evolution of computers, we shall now emphasize recent developments. We do not intend to present a lot of potentially boring statistics, but at this point it does seem that quantitative information is one of the best ways to give the reader a perspective concerning the rapid changes that have recently occurred in the industry.

Table 1.1 presents a contemporary and estimated near-future overview of computer technology as measured by various economic and engineering parameters. The first parameter, computer generations, is a popular means for indicating the major periods of development. To a large extent, these intervals are determined by the available electronic components. (See the second line in the table.) There are other factors, however, which are unique to the various generations. One of these is programming languages. During the first generation, only elementary languages were available, but in later generations very sophisticated programming systems have been employed. (Some of these will be considered in Chapter 3.) An extensive discussion of the various computer generations is found in the literature.[3]

Observe from the table that during the decade from 1960 to 1970 there was approximately an order of magnitude increase in both the economic importance of computers (e.g., note the increased value of computers shipped) and their performance (e.g., note the decrease in logic delay). Moreover, these were years of generally rising manufacturing costs; yet the prices for multiplications, minicomputers, and memory all showed very sharp decreases.

Another aspect of the progress in computer hardware can be obtained from the photograph in Figure 1.5, which shows comparable circuitry in each of the first three computer generations. The original vacuum-tube circuitry, a transistor version, and an integrated-circuit revsion are all easily identified. Actually, the reduction in size is compounded in that there is a corresponding change in power supplies, cooling, and other support functions.

Another facet of computer technology is concerned with the corporations which constitute the industry. Four companies, IBM, Honeywell, Sperry Rand, and Control Data, with sale volume in that order, do over 85% of the business. Moreover, IBM itself has close to 75% of the industry's income. This company employs approximately 350,000 people (for comparison, General Electric employs about 400,000 in all divisions), and it has had a 390% increase in gross income since 1959 (the gross national product increased 88% in the same period). Perhaps even more important, IBM is the top company of the world from the standpoint of the market value of its stock (approximately $40 billion as compared to about $25 billion for American Telephone and Telegraph, which is in second place). One of IBM's largest computers, the System 370/168, is shown in Figure 1.6.