DESIGN OF DIGITAL COMPUTERS

Hans W. Gschwind   Edward J. McCluskey

Springer-Verlag
Berlin Heidelberg New York

Design of Digital Computers

An Introduction

SECOND EDITION

Hans W. Gschwind
Edward J. McCluskey

Springer-Verlag
Berlin Heidelberg New York
1975

Hans W. Gschwind
Standard Elektrik Lorenz AG
Hellmuth-Hirth Strasse 42
Postfach 4007 49 D-7 Stuttgart 40
West Germany

Edward J. McCluskey
Director
Digital Systems Laboratory
Departments of Electrical
Engineering and Computer Science
Stanford, California 94305
USA

Editors

F. L. Bauer
Mathematisches Institut der
Technischen Hochschule
8000 München
Arcisstrasse 21
West Germany

David Gries
Cornell University
Department of Computer Science
Upson Hall
Ithaca, New York 14859
USA

AMS Subject Classification (1970): 68-01, 94-01, 94A20
(C. R.) Computing Reviews Classifications: 1.1, 6.1, 6.3

Library of Congress Cataloging in Publication Data

Gschwind, Hans W
Design of digital computers.

(Texts and monographs in computer science)
Includes indexes.
1. Electronic digital computers—Design and construction. I. McCluskey, E. J., joint author.
II. Title.
TK7888.3.G72 1975 621.3819'58'2 74-31174

Printed in the United States of America.

ISBN 0-387-06915-1 Springer-Verlag New York Heidelberg Berlin
ISBN 3-540-06915-1 Springer-Verlag Berlin Heidelberg New York

Texts and Monographs in Computer Science

F. L. Bauer
David Gries
editors

Preface

I have been using the first edition of this book as a text for a number of years. This was in a Stanford University first-year graduate course that is taken by students from Electrical Engineering or Computer Science who are interested in computer organization. Because computer technology has been changing so rapidly, it became necessary to supplement the text with additional readings. My colleagues and I examined many newly-published books for possible use as texts. We found no book with the same excellent choice of topics and thorough coverage as Dr. Gschwind's first edition.

Springer-Verlag's request that I prepare a second edition of this book came at a time when I had many other projects underway. Before I decided whether to take on the project of preparing a revision, I asked many of my students for their opinions of Dr. Gschwind's first edition. Even I was surprised by the enthusiasm that this rather skeptical and critical group of students displayed for the book. It was this enthusiasm that convinced me of the value and importance of preparing the revision.

Most of the changes and additions that I have made are concerned with the primary role played by the integrated circuit in contemporary computers and other digital systems. Thus Chapter 4 has been entirely rewritten. It now presents a comprehensive discussion of the various technologies used to implement digital integrated circuits. Every attempt was made in developing this presentation not to require a background in electric-circuit theory. This chapter has been taught successfully to computer science students with no electrical engineering or physics background. This is possible not because of an avoidance of the important circuit phenomena but by carefully developing mathematical models for the integrated circuit diode and transistors. These models incorporate those parameters which are significant in explaining the performance of the digital circuits. I believe that it has thus been possible to have a presentation which while accessible to the computer scientist is still complete enough for the electrical engineer.

When preparing the material on flip-flops, I discovered that there was a great deal of confusion about the essential difference between edge-triggered and master-slave flip-flops. The discussions of these flip-flops in the manufacturers' literature, in books and periodicals were all somewhat confused. I believe that after many discussions with friends in integrated circuit companies as well as colleagues in computer engineering I have been able to present in Chapter 5 a succinct summary of the essential features of integrated circuit flip-flops.

The material on counter circuits in Chapter 6 has been entirely rewritten to bring it up to date with contemporary integrated circuit techniques. Chapter 8 has a major section on integrated circuit memories added. Also the material on microprogrammed control units has been incorporated into Chapter 8 and combined with the discussion of other techniques of designing control units.

The new material included in this revision has been used in classes both at Stanford University and the University of Illinois. I have benefited from helpful comments from students on this material.

Many people have helped in the preparation of this manuscript. Mrs. Susan Jordan did much of the typing and also did a beautiful job of drawing most of the figures. The early typing was done by Mrs. Patricia Fleming. Copy editing, emergency typing, reference citation preparation, and proofreading were done by Mrs. Sally Burns, now Mrs. Sally McCluskey.

Many of my students and former students commented on the manuscript and helped with proofreading. John Wakerly, Kenneth Parker, John Shedletsky and Rodolfo Betancourt all helped with the reading of the galley proofs.

February 1975 EDWARD J. MCCLUSKEY

Design of Digital Computers

Contents

1. Introduction

"Computers" have attracted general interest only rather recently although computing devices have been known for a long time. The Antikythera mechanism, supposedly used by ancient Greeks to determine the motions of the stars and planets [1], the astrolabes of the middle ages [2], and Pascal's calculator [3], are only a few examples of early computational devices. However, the present usage of the term "computer" includes neither those relatively primitive (though certainly effective) aids for computation, nor later developments like the slide rule, the planimeter, or the desk calculator. What we mean nowadays by a computer is a machine which performs a computation automatically and without human intervention, once it is set up for a specific problem. If we want to emphasize this distinction, we speak of automatic computers as opposed to calculators or computing devices.

The present use of the term "computer" has a second connotation. It usually refers to an electronic device, although there have been (and still are) automatic computers which operate mechanically or electromechanically. There are mainly two reasons for this close association between electronics and modern computers: no known principle other than electronics allows a machine to attain the speeds of which modern computers are capable; and no other principle permits a design of comparable convenience.

Even though practically all modern computers operate electronically, there are several distinct types of machines. Here, we do not mean differences concerning circuit elements such as tubes or transistors, but basic differences in the design philosophy. The most characteristic distinction is probably the analog or digital nature of a computer.

An analog computer represents the values which it uses in its calculation by physical quantities. The slide rule, which is an analog device (although, of course, no computer according to our definition), uses the physical quantity "length" to represent computational values. An electronic analog computer uses voltages as convenient analog quantities (higher voltages for larger values, lower voltages for smaller values, etc.) In contrast, a digital computer employs numbers, as we usually do in paper and pencil calculations. Numerical values are represented by the presence or absence of electric potentials or pulses on certain lines. The magnitude of these potentials or pulses is of no particular significance, as long as it is adequate for the fault-free operation of the computer. Of course, both basic representations have their merits and their disadvantages.

Analog Computers are of relatively uncomplicated design. It is quite feasible to build small and inexpensive machines. Moreover, problems put on an analog computer usually are simulated by an electronic network of resistors, capacitors, amplifiers, etc., which has an intelligible relationship to the problem to be solved [4]. On the other hand, the electronic model usually comes only to within about 1% or .1% of a true representation of the actual problem. Even though this inherent error is of no importance for many problems, there are calculations in which it cannot be tolerated. Furthermore, there are several types of problems which, due to the nature of its design, cannot be solved by an analog computer[1].

Digital Computers are relatively complex machines. In many instances, it is difficult for an expert to recognize from the program alone even the type of problem to be solved. However, digital computers have the great advantage that they can solve practically all problems which can be stated in mathematical language. Their accuracy is not limited by the operating principle, but only by practical considerations. Furthermore, they can be employed to translate their own internal language into very concise and intelligible statements or, conversely, interpret instructions given in almost everyday language for their own use.

In addition to analog and digital computers, there are a few computer types which attempt to combine the advantages of both principles. The *Digital Differential Analyzer,* similar to an analog computer, represents problems by a network of units (the integrators) but, like a digital computer, uses numbers to represent computational values [5]. In *Hybrid Computers,* analog computations are combined with digital computations [6].

Of these four types of computers, only the digital computer will be considered here[2]. The topics may be roughly divided into four categories.

[1] For instance, inventory control, bookkeeping, playing of mathematical games, or, perhaps, finding all prime numbers between 0 and 10^6.

[2] It is, however, worthwhile to note that some of the indicated basic design techniques may be applied to any digital equipment, including that of digital differential analyzers and hybrid computers, and that the organization of these latter two types of computers is at least sketched in order to provide a reference against which the organization of digital computers may be viewed.

Chapters 2 and 3 contain fundamental information on number systems and Boolean algebra. The detail included provides more than a prerequisite for the understanding of the then following material. Chapters 3, 4, 5, and 6 are concerned with individual components and circuits which constitute the computer hardware. Chapters 7, 8, and 9 are devoted to the organization of computers. Chapter 10 contains miscellaneous topics not included elsewhere.

Problem 1 (Voluntary): You are provided a desk calculator and an operator for it. The operator can execute only simple instructions such as "add the value in column 7 to the value in column 5" or "copy down result in column 9". Try to devise a set of instructions and a worksheet for the operator so that he can calculate the value of sin x for any given x by the approximation:

$$\sin x \approx x - \frac{x^3}{6} + \frac{x^5}{120}$$

References

1. Derek, and de Solla Price: An Ancient Greek Computer, Scientific American, vol. 200, No. 6, pp. 60–67. June 1959.
2. Schroeder W.: Practical Astronomy. London: T. Werner Laurie Ltd. 1956.
3. Willers A.: Mathematische Instrumente. Munich: R. Oldenbourg. 1943.
4. Korn, and Korn: Electronic Analog Computers. New York: McGraw-Hill. 1952.
5. Forbes G. F.: Digital Differential Analyzers. Private Print: 13745 Eldridge Ave, Sylmar, California. 1957.

   Handel P. von: Electronic Computers. Vienna: Springer. Englewood Cliffs: Prentice Hall. 1961.
6. Burns M. C.: High Speed Hybrid Computer, National Symposium on Telemetering, San Francisco. 1959.

   Birkel G., Jr.: Mathematical Approach to Hybrid Computing, National Symposium on Telemetering, San Francisco. 1959.

   Truitt T. D.: Hybrid Computation, AFIPS Conference Proceedings, vol. 25, pp. 249–269, Spring Joint Computer Conference. 1964.

Selected Bibliography

A history of early digital computers is contained in:

Staff of the Computation Laboratory Harvard, A Manual of Operation for the Automatic Sequence Controlled Calculator. Cambridge. 1946.

Eames, and Fleck: A Computer Perspective. Cambridge, Mass.: Harvard University Press. 1973.

2. Number Systems and Number Representations

The familiar decimal system is by no means the only possible number system. Considered impartially, it merely constitutes one among possible and practical systems which became propagated, probably for the sole reason that human beings happen to have ten fingers. The Mayas used the vigesimal number system (based upon 20, i.e. fingers and toes) [1] and even in our day, there are some endeavors to introduce the duodecimal system (based on 12) for general use [2]. Since computers are not bound by tradition and since the decimal system has no unique merits, the designer of a computer is free to select that number system which suits his purpose best.

2.1 Counting in Unconventional Number Systems

Before we set out on unfamiliar ground, let us shortly review the decimal number system. Any decimal number is made up of the ten symbols: 0, 1, 2, . . . 9. When we count, we use these symbols consecutively: 0, 1, 2, . . . 9. Then, if we have exhausted all available symbols, we place the symbol 1 in a new position and repeat the cycle in the old position: 10, 11, . . . 19. If we run out of symbols again, we increase the digit in the second position and repeat the cycle in the first position: 20, 21, . . . 29, etc. If we have no more symbols for the second position, we create a third position: 100, 101, and so on.

Counting in a different number system follows the same procedure. Let us count in the ternary system (base 3). We have only three symbols: 0, 1, and 2. We proceed as follows: 0, 1, 2. Then having no other symbols for this position, we continue: 10, 11, 12. Running out of symbols again, we write: 20, 21, 22. Having no more symbols for the second position, we create a third position: 100, 101, 102, 110, 111, 112, 120, 121, 122, 200, 201 and so on.

Problem 1: Count in the binary system (base 2) and in the duodecimal system (base 12) up to the equivalent of the decimal number 25.

Use the letters T and E as symbols for ten and eleven in the duodecimal system.

Problem 2: Try to state some advantages and disadvantages which the duodecimal system might have over the decimal system for computers and for everyday calculations.

2.2. Arithmetic Operations in Unconventional Number Systems

We can perform calculations in other number systems equally well as in the decimal system, once we are familiar with a few simple rules. For arithmetic operations in the decimal system, we (mentally) use the addition and multiplication tables reproduced below.

Table 2.1. *Decimal Addition Table*

+	0	1	2	3	4	5	6	7	8	9
0	0	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9	10
2	2	3	4	5	6	7	8	9	10	11
3	3	4	5	6	7	8	9	10	11	12
4	4	5	6	7	8	9	10	11	12	13
5	5	6	7	8	9	10	11	12	13	14
6	6	7	8	9	10	11	12	13	14	15
7	7	8	9	10	11	12	13	14	15	16
8	8	9	10	11	12	13	14	15	16	17
9	9	10	11	12	13	14	15	16	17	18

Table 2.2. *Decimal Multiplication Table*

×	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9
2	0	2	4	6	8	10	12	14	16	18
3	0	3	6	9	12	15	18	21	24	27
4	0	4	8	12	16	20	24	28	32	36
5	0	5	10	15	20	25	30	35	40	45
6	0	6	12	18	24	30	36	42	48	54
7	0	7	14	21	28	35	42	49	56	63
8	0	8	16	24	32	40	48	56	64	72
9	0	9	18	27	36	45	54	63	72	81

Table 2.3. *Ternary Addition and Multiplication Tables*

+	0	1	2
0	0	1	2
1	1	2	10
2	2	10	11

×	0	1	2
0	0	0	0
1	0	1	2
2	0	2	11

Let us construct corresponding tables for, let us say, the ternary system. Having only three symbols, we will obtain nine entries. Instead of the decimal symbols for three and four, we will show their ternary equivalents 10 and 11. (See Table 2.3).

We can use these tables for calculations in the ternary system in the same manner as we use the decimal tables for computations in the decimal system. Suppose we want to add the two ternary numbers 1021220 and 210121. The computation is given below:

```
  1 1   1 1
  1 0 2 1 2 2 0
+   2 1 0 1 2 1
---------------
  2 0 0 2 1 1 1
```

The carries to be brought forward are indicated in the top line. Similarly, for the product of the two ternary numbers 1120 and 12, we obtain:

```
      1 1 2 0
×         1 2
-------------
    1 0 0 1 0
    1 1 2 0
-------------
    2 1 2 1 0
```

The simplest addition and multiplication tables are obtained for the binary system:

Table 2.4. *Binary Addition and Multiplication Tables*

+	0	1
0	0	1
1	1	10

×	0	1
0	0	0
1	0	1

The simplicity of these tables is perhaps one of the reasons why the binary number system is so attractive to computer designers.

From now on, we will indicate the base of a number by an appropriate index if the base is not apparent from the context. For instance, a number like 453_8 shall indicate an octal number (base 8).

Problem 3: Construct the addition and multiplication tables for the quinary (base 5) and octal (base 8) number systems. Be sure to make all entries in the appropriate number system.

Problem 4: Construct the addition tables for the duodecimal (base 12) and the hexadecimal (base 16) systems. Use the letters *T* and *E* as symbols for ten and eleven in the duodecimal system and the letters *A, B, C, D, E, F* as symbols for numbers from ten to fifteen for the hexadecimal system.

Problem 5: Perform the following arithmetic operations:

a) $10111_2 + 1101_2$
b) $11010_2 - 10110_2$
c) $101101_2 \times 1011_2$
d) $11011_2 \div 11_2$
e) $2431_5 + 132_5$
f) $324_5 \times 14_5$
g) $6327_8 + 4530_8$
h) $124_8 - 76_8$
i) $1256_8 \times 27_8$

Check your computations by converting these problems and their results to the decimal system after you have studied paragraph 2.3.

2.3. Conversions

As long as there is more than one number system in use, it will be necessary to convert numbers from one system to another. Such a conversion is required if we want to insert decimal numbers into a binary computer, or vice versa, if we want to interpret results computed by such a machine. If we are to do this conversion ourselves, we prefer to perform the required arithmetic in the familiar decimal system. If the computer performs the conversion, an algorithm in its number system is preferable.

Each position in a decimal number like 2536 has a certain weight associated with it. The digit 2 in the above number represents, for example, two thousand or its position has the weight 10^3. Writing the number 2536 in longhand, we have:

$$2536_{10} = 2 \times 10^3 + 5 \times 10^2 + 3 \times 10^1 + 6 \times 10^0$$

An arbitrary decimal number has the form:

$$N_{10} = \cdots d_3 \times 10^3 + d_2 \times 10^2 + d_1 \times 10^1 + d_0 \times 10^0 + d_{-1} \times 10^{-1} + \cdots \quad (2.1)$$

A number written in a system other than decimal has the same general structure; only the weights will be different. For an arbitrary number written in the octal system, we obtain for instance:

$$N_8 = \cdots C_3 \times 8^3 + C_2 \times 8^2 + C_1 \times 8^1 + C_0 \times 8^0 + C_{-1} \times 8^{-1} + C_{-2} \times 8^{-2} + \cdots \quad (2.2)$$

The coefficients C_n are octal integers ranging from 0_8 to 7_8.

Conversion formulae derive the coefficients in one number system (e.g. Equation 2.1) from the coefficients of another number system (e.g. Equation 2.2). Since the procedures are different for different conversions, we will consider one case at a time.

2.3.1. Conversion of Integers

Let us start with a specific example. Suppose we want to convert the number 3964_{10} to the octal system. This number is an integer in the decimal system and consequently also an integer in the octal system. (We can derive it by counting "units".) According to Equation (2.2) we can write in general terms:

$$3964_{10} = \cdots C_3 \times 8^3 + C_2 \times 8^2 + C_1 \times 8^1 + C_0 \times 8^0 \quad (2.3)$$

All C's are positive integers smaller than 8, but not yet determined.

Suppose we split the right-hand side of Equation (2.3) into two parts:

$$3964_{10} = (\cdots C_3 \times 8^2 + C_2 \times 8^1 + C_1) \times 8 + C_0 \quad (2.4)$$

The first term, apparently, is part of our original number which is divisible by 8 (the integral part of the quotient $3964_{10} \div 8$), whereas the term C_0 is that part of the original number which is not divisible by 8 (the remainder of the quotient $3964_{10} \div 8$).

If we divide 3964_{10} by 8, we obtain:

$$3964_{10} \div 8 = 495 + 4/8$$

We can therefore write:

$$3964_{10} = 495 \times 8 + 4 \quad (2.5)$$

Comparing (2.4) and (2.5), we find $C_0 = 4$, or we can write

$$3964_{10} = 495_{10} \times 8^1 + 4_8 \times 8^0 \quad (2.6)$$