microcomputer design and applications

EDITED BY

Samuel C. Lee

microcomputer
design and applications

ACADEMIC PRESS RAPID MANUSCRIPT REPRODUCTION

microcomputer design and applications

EDITED BY

Samuel C. Lee
School of Electrical Engineering
and Computing Sciences
University of Oklahoma
Norman, Oklahoma

ACADEMIC PRESS New York San Francisco London 1977
A Subsidiary of Harcourt Brace Jovanovich, Publishers

COPYRIGHT © 1977, BY ACADEMIC PRESS, INC.
ALL RIGHTS RESERVED.
NO PART OF THIS PUBLICATION MAY BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS, ELECTRONIC OR MECHANICAL, INCLUDING PHOTOCOPY, RECORDING, OR ANY INFORMATION STORAGE AND RETRIEVAL SYSTEM, WITHOUT PERMISSION IN WRITING FROM THE PUBLISHER.

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
ACADEMIC PRESS, INC. (LONDON) LTD.
24/28 Oval Road, London NW1

Main entry under title:

Microcomputer design and applications.

1. Microcomputers—Congresses. I. Lee, Samuel C., Date
QA76.5.M5217 001.6'4'04 77-8612
ISBN 0-12-442350-7

PRINTED IN THE UNITED STATES OF AMERICA

Contents

List of Contributors

J. BRADLEY Texas Instruments, Inc., 12201 Southwest Freeway, Stafford, Texas 77477

ROBIN L. BROWN Design Automation, Texas Instruments, Inc., Dallas Texas 75260

ELAINE B. BUBA Product Programming and Development, General Dynamics, Fort Worth, Texas 76101

S. CARR Union College, Schenectady, New York 12308

Y. P. CHIEN Department of Information Engineering, University of Illinois at Chicago Circle, Box 4348, Chicago, Illinois 60680

H. M. DREIZEN Department of Information Engineering, University of Illinois at Chicago Circle, Box 4348, Chicago, Illinois 60680

GREGG DUMAS Fairchild Micro System, 1725 Technology Drive, San Jose, California 95110

ALBERT D. EDGAR School of Electrical Engineering and Computing Sciences, University of Oklahoma, Norman, Oklahoma 73019

P. DAVID FISHER Michigan State University, East Lansing, Michigan 48824

A. FRITZSON Union College, Schenectady, New York 12308

ROBERT M. GLORIOSO Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, Massachusetts 01002

DAVID R. HADDEN, JR. US Army Electronics Command, Fort Monmouth, New Jersey 07703

THOMAS M. HOULIHAN Department of Mechanical Engineering, Naval Postgraduate School, Monterey, California 93940

THOMAS P. HUGHES US Army Electronics Command, Fort Monmouth, New Jersey 07703

L. HULL Union College, Schenectady, New York 12308

ALAN L. JONES Michigan State University, East Lansing, Michigan 48824

PAUL N. KIZAKEVICH Veterans Administration Hospital, 1201 N. W. 16th Street, Miami, Florida 33125

REX C. KLOPFENSTEIN The MITRE Corporation, METREX Division, 7925 Westpark Drive, McLean, Virginia 22101

R. A. KOBYLINSKI Department of Electrical Engineering, University of Missouri—Rolla, Rolla, Missouri 65401

SAMUEL C. LEE School of Electrical Engineering and Computing Sciences, University of Oklahoma, Norman, Oklahoma 73019

SIGURD L. LILLEVIK Department of Electrical Engineering and Systems Science, Michigan State University, East Lansing, Michigan 48824

VINCENT C. NEGRO US Energy Research and Development Administration, Health and Safety Laboratory, 376 Hudson Street, New York, New York 10014

D. OGDEN Texas Instruments, Inc., 12201 Southwest Freeway, Stafford, Texas 77477

JOHN R. PLUNKETT Miramar Naval Air Station, San Diego, California 92135

THOMAS REISER The Cooper Union for the Advancement of Science and Art, 51 Astor Place, New York, New York 10003

CHARLE' R. RUPP General Electric Co., Pittsfield, Massachusetts 01201

DWIGHT H. SAWIN III US Army Electronics Command, Fort Monmouth, New Jersey 07703

KURT A. SCHEMBER Computer Science and Engineering, University of Texas at Arlington, Arlington, Texas 76019

EDWARD L. SHREVE School of Electrical Engineering, Oklahoma State University, Stillwater, Oklahoma 74074

ANDREW SIMMONS Veterans Administration Hospital, 1201 N. W. 16th Street, Miami, Florida 33125

P. D. STIGALL Department of Electrical Engineering, University of Missouri—Rolla, Rolla, Missouri 65401

B. G. STRAIT Los Alamos Scientific Laboratory, P. O. Box 1663, Los Alamos, New Mexico 87545

J. M. TAYLOR Department of Electrical Engineering, University of Missouri—Rolla, Rolla, Missouri 65401

J. W. THOMAS Texas Instruments, Inc., 12201 Southwest Freeway, Stafford, Texas 77477

M. E. THUOT Los Alamos Scientific Laboratory, P. O. Box 1663, Los Alamos, New Mexico 87545

MONTE TULL Western Electric Company, Inc., 7725 W. Reno, Oklahoma City, Oklahoma 73108

T. C. WERNER Chemistry Department, Union College, Schenectady, New York 12308

DAVID E. WEST Magnetic Peripherals, Inc., 4000 N. W. 39th Street, Oklahoma City, Oklahoma 73112

G. WILLIAMS Department of Computer Science, Union College, Schenectady, New York 12308

R. E. ZIEMER Department of Electrical Engineering, University of Missouri—Rolla, Rolla, Missouri 65401

Preface

The advent of the microcomputer marks the beginning of a new computer revolution in our technological society. The widespread application of microcomputers, from process controls to medical instrumentations, from computer peripherals to home entertainment sets, indicates the revolution is well underway. Due to these rapid developments in microcomputer design and applications, there is a general need for a book giving an overview of this area. *Microcomputer Design and Applications* was born out of this need.

This book, containing 16 carefully selected papers, is divided into two parts: Part 1, Microcomputer Design (papers 1–5); Part 2, Microcomputer Applications (papers 6–16). A brief description of the papers follows.

The first paper introduces a new number system and supporting computational algorithms, called *Focus,* which is especially useful for microcomputer control and digital signal processing. Microcomputer implementations of Focus computational algorithms are included. In the second paper, a class of computer architectures is presented having the property that the interconnection network of dedicated function processors that make up a computer is generated from a so-called *control grammar.* A design approach for multiple-processor computers is described wherein a definition of the desired capabilities of a computer using the concept of a control grammar may be used to generate a class of multiple-processor architectures. The approach potentially may result in an automated methodology for the development of high-performance computers based upon microprocessors. The third paper presents an integrated technical and management-based approach for developing microprocessor software. "LSI software" is a by-product of this approach. LSI software methodology is a promising concept for developing ROM-based systems.

File structures for a small-scale database system designed for microprocessor implementation are discussed in the fourth paper. The formulation and evaluation of file structures for a "typical" microcomputer/flopping disk system is presented. This includes performance evaluation of both primary and secondary file structures. In the fifth paper, several problems of software development for microprocessor-based computer systems are addressed. The proposed solution is to specify a high-level, machine-oriented, structured programming language suitable for general microprocessors and to implement a portable compiler for this language.

As examples of microprocessor/microcomputer applications, a microprocessor stepping-motor controller and a microcomputer CRT controller are presented in the sixth and the seventh papers, respectively. A data-logging system is described in the eighth paper; the system uses a microcomputer to implement four independent data channels, a time-of-year channel, and an input-output write channel. Three microcomputer-based data acquisition systems are presented in the next three papers. The construction and operation of a fully automated microprogrammable data acquisition and control system with application to the sampling, averaging, and recording of atmospheric meteorological data is reported in the ninth paper. The tenth paper describes the design and operation of a microprocessor-based system that performs three functions: (1) instrument control, (2) data acquisition, and (3) time-averaging processing. The eleventh paper presents a microcomputer data acquisition system that demonstrates the feasibility of using a microprocessor to analyze transionospheric communication channels. The microcomputer approach leads to several benefits including portability, field data analysis, and fewer resources involved in data acquisition.

To demonstrate the wide range of microprocessor/microcomputer applications, three examples are presented in the twelfth, the thirteenth, and the fourteenth papers. They are a microcomputer-based fire control radar data processor, a battery-powered CMOS microprocessor-based instrument for agriculture, and a microprocessor-based data collection system for transit buses.

Two multi-microprocessor/multi-microcomputer applications are presented in the last two papers. The fifteenth paper describes the use of two identical 16-bit microcomputers to increase throughput by performing simultaneous execution of programs. The application is a handwriting recognition machine intended for data entry applications. A multiple processor system that performs parallel data acquisition, analysis, and display is presented in the last paper.

The consistently outstanding work of the staff of Academic Press has removed much of the onus for the authors and editor alike in the preparation of this book. Finally, I would like to express my thanks to the authors of the individual papers, who have worked so competently and conscientiously to provide this extremely important collection.

microcomputer
design and applications

This page intentionally left blank

1

FOCUS MICROCOMPUTER NUMBER SYSTEM

Samuel C. Lee
Albert D. Edgar
The University of Oklahoma

Focus is a number system and supporting computational algorithms especially useful for microcomputer control and other signal processing applications. Focus has the wide ranging character of floating point numbers with a uniformity of state distributions that give focus more than a two-fold accuracy advantage over an equal word length floating-point system. Focus computations are typically five times faster than single precision fixed-point or integer arithmetic for a mixture of operations, comparable with hardware arithmetic in speed for many applications. Algorithms for 8-bit and 16-bit implementations of focus are included.

I. INTRODUCTION

A digital control system should respond qualitatively strong to gross errors between the output and control signal, but should respond quantitatively delicate as equilibrium is approached. The latter requirement often demands a minimum of 12 to 16-bit fixed-point systems to avoid chatter caused by quantum jumping near quiescence, even though most of the resulting thousands of states are wasted in providing fractional parts-per-thousand control of the corrective force when there is a large error in the system being controlled. A much more efficient number system would concentrate available states near zero in analogy to the "virtual ground" of an operational amplifier or in analogy to the human eye which concentrates available resolution near a "center of focus" while retaining cognizance of distant states. The number system presented does allocate available states in this fashion, thus suggesting its name.

If there is time, a computer has an inherent ability to respond to the environment with an heuristic and complex nature that cannot be matched in analog form. The number system presented provides the speed required for a clearer focus of the complexities of the system environment and its control, thus again suggesting its name.

As an introduction to the focus system, its abilities are now specified using numbers and statements that will be supported as the concept is developed.

An 8-bit focus word provides the range and resolution in a fairly large area around and including zero of a $15\frac{7}{8}$ -bit fixed-point word. Compared to the logarithmically uniform focus, a floating-point representation seems "out of focus" with typically one bit less efficiency based on RMS fractional error, two bits less based on peak fractional error, and further accuracy losses in computation. With the focus coding a single 8-bit byte is for the first time in serious contention for non-trivial applications, providing bipolar operation with a $2\frac{1}{2}$ % RMS fractional error over a range of 60,000 to 1 either side of zero. For comparison, 8-bit focus gives less audible distortion of a signal than would transcription on a cassette tape recorder. A 2-byte 16-bit focus word fulfills the requirements of most precision signal processing applications, providing 0.13% RMS fractional error over a range of 64 orders of magnitiude for positive and negative numbers.

Unlike floating-point software that gives accuracy at the expense of speed, the special feature of focus is accuracy with speed. Compared with a uniform mixture of add, subtract, multiply, and divide, and with 8-bit fixed-point computations performed on a popular microprocessor, 8-bit focus executes 5 times faster. This is not a fair number however because the fixed-point algorithms used for comparison do not fixup overflow, and the fixed-point software multiply and divide are only for positive numbers. When overflow fixup and 4 quadrant ability are removed also from the focus program, the 8-bit focus executes 10 times faster than the 8-bit fixed-point. A set of 16-bit focus algorithms with overflow fixup and 4 quadrant ability executes 3 times faster than the 8-bit fixed-point comparison but provides the nature and accuracy of an 18-bit floating point system.

II. REPRESENTING A NUMBER IN THE FOCUS SYSTEM

A. Binary Representation

Focus is a logarithmic number coding in which, like floating-point, an exponent is stored, but unlike floating-point, no mantissa is given. Rather the exponent is a fixed-point number with a fractional part. Both the number and the exponent have signs which could be represented in sign-magnitude, offset binary, or one's or two's complement. The combination of sign-magnitude for the number and offset binary for the exponent sign is found to be optimum.

Sign bit
"0" if the number is positive
"1" if the number is negative

$$|\boxed{B}|\ |\boxed{C}|\ |\boxed{D}|\ |\boxed{E}|\ .\ |\boxed{F}|\ |\boxed{G}|\ |\boxed{H}|$$

$|\boxed{A}| \cdot 2$

Remaining 7 bits hold the base-2 logarithm of the absolute value of the number plus an offset of 1000.000 with the binary point fixed as shown.

Fig. 1. Preferred realization for an 8-bit focus word.

An example of a focus realization is given in figure 1. The formatting is the preferred realization for an 8-bit word, and will be used in illustrating the computational algorithms. Sample focus words are given:

$$0\ 1000.000 = +2^{0} = 1$$

$$1\ 1001.000 = -2^{1} = -2$$

$$0\ 0111.000 = +2^{-1} = \frac{1}{2}$$

$$0\ 1000.101 = +2^{5/8} \simeq 1\frac{1}{2}$$

$$0\ 1111.111 = +2^{7\frac{7}{8}} \simeq 235$$

$$1\ 0000.000 = -2^{-8} \simeq -.004$$

B. Analog Encoding and Decoding for Focus

The simplified schematic shown in figure 2 interfaces an analog signal with focus binary formatting. Translation of the signal while in analog form allows the characteristic resolution and range of focus to be retained with fixed-point digital converters of reasonable bit size. Thus although the proposed 8-bit focus spans nearly five orders of magnitude, only 7-bit fixed-point converters are needed. 12-bit fixed-point converters service the proposed 16-bit focus format over a continuous bipolar range of 1 microvolt to 100 volts.

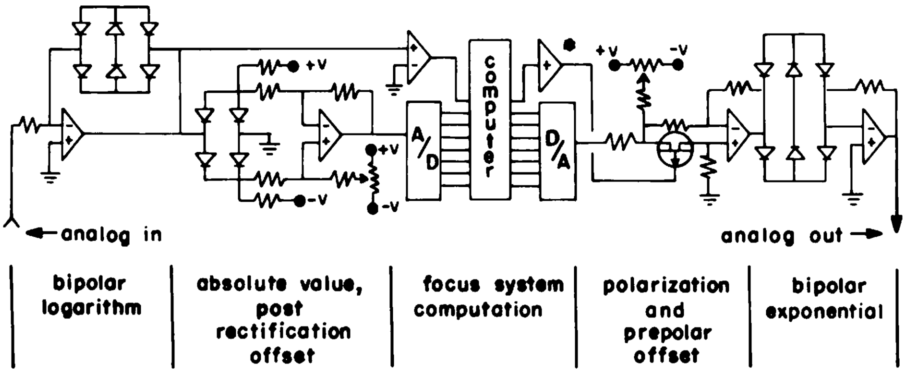

Fig. 2. An analog circuit for coding-decoding of focus numbers.

Because focus has special accuracy near virtual ground with special capacity for wide dynamic excursions with lesser accuracy, it is often desirable to perform subtraction of differential input signals in analog form and to perform output integration in analog form, both in analogy to the functioning of a differential input internally stabilized operational amplifier. Similarly in audio processing frequency pre-emphasis can be used with a minimum of concern for overranging. A dynamic expander-contractor is not needed as this feature is inherent in the logarithmic focus coding.

III. ARITHMETIC OPERATIONS IN THE FOCUS SYSTEM

A. Addition and Subtraction

The vital feature of the focus number system is the ability to perform general calculations and to perform them rapidly. This hinges on the ability to perform addition directly in the logarithmic domain.

A theoretically possible but impractical method of addition would be to use a two dimensional lookup table. 16-bit addition using this method would require over four billion words of memory. Another theoretical possibility would be to use two 64K word arrays to translate back and forth between the linear and logarithmic domains. However such a system would execute slowly and increase computational error with two stages of roundoff.

The focus addition/subtraction algorithm does use a lookup table, but requires only a little over 2000 8-bit <u>half</u>

word bytes for the preferred <u>16-bit</u> focus realization, and 256 bytes for the preferred 8-bit system. Execution is extremely fast, requiring only a single memory reference, sign test, and two binary additions when adding two positive focus numbers. All answers materialize rounded to the nearest available state for the minimum possible computation error.

The basic focus addition algorithm for two positive numbers is now presented. Let X, Y, and Z be the focus coded numbers representing the real world values X', Y', and Z', i.e.:

$$X = \log_2(X') + \text{offset}$$

$$Y = \log_2(Y') + \text{offset}$$

and the sum

$$Z = \log_2(X' + Y') + \text{offset}$$

We wish to find the focus coded sum Z from X and Y with no reference to the real world values X' and Y'. To do this, the following operation is performed:

$$Z = X + F(X - Y)$$

The addition and subtraction are standard binary operations. F(N) is a one dimensional lookup table which is precalculated according to the formula derived below:

$$Z = \log_2(X' + Y') = X + F(X - Y)$$

$$X' + Y' = 2^{X + F(X - Y)}$$

$$2^X + 2^Y = 2^X \cdot 2^{F(X - Y)}$$

$$1 + \frac{2^Y}{2^X} = 2^{F(X - Y)}$$

$$\log_2(1 + 2^{Y-X}) = F(X - Y)$$

$$\log_2(1 + 2^{-N}) = F(N)$$

The offsets in X and Y cancel in the subtraction, leaving N as simple fixed point, which may be arranged to always be positive. However, when referring to a table entry, the subscript N will be the integer value derived by ignoring the binary point. The value in storage, F(N), is always positive, and considered fixed-point. Subtraction, as with 4 quadrant addition, uses a complementary table $F(N) = \log_2(1 - 2^{-N})$.