INTRODUCTION TO COMPUTER-ASSISTED EXPERIMENTATION

Kenneth L. Ratzlaff

INTRODUCTION TO COMPUTER-ASSISTED EXPERIMENTATION

Kenneth L. Ratzlaff
Director, Instrumentation Design Laboratory
Department of Chemistry
University of Kansas
Lawrence, Kansas

A WILEY-INTERSCIENCE PUBLICATION

JOHN WILEY & SONS
NEW YORK CHICHESTER BRISBANE TORONTO SINGAPORE

Library of Congress Cataloging in Publication Data:

Ratzlaff, Kenneth L.
Introduction to computer-assisted experimentation.

"A Wiley-Interscience Publication."
Includes index.
1. Physical laboratories—Data processing.
2. Microcomputers. I. Title.

QC51.A1R38 1987 502.8'5 86-19011
ISBN 0-471-86525-7

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

Preface

During the past several years, a profound change has taken place in the scientific laboratories of universities, industries, government labs, and other research centers. The personal computer has arrived. This machine has the computing power rivaling that of many mainframes in centralized facilities a little more than a decade ago. However, the cost is much *less* than that of a typical scientific instrument. Even more important, it is becoming increasingly "friendly." No longer does computer operation require specialized training. The mystique of the "computer jock" is finally disappearing.

Computers demonstrate a wide range of capabilities. On the one hand, they can be and have been used for our detriment. Computers are popularly known for their ability to act on behalf of "Big Brother." It is sobering also to note that a primary impetus in their development has been support of the tools of war; indeed, some of today's most challenging computer problems exist in technologies which threaten world peace, such as the "Star Wars" initiative.

On the other hand, computers also are aids to efforts that can help bring about peace and improvements in mankind's lot. They are indispensable aids to modern fundamental research; educators find them to be valuable assistants; and they can take over much of the drudgery associated with repetitive tasks. The computer is, for most scientists, a positive tool for research with positive benefits. It is hoped that this book will assist those studies.

Despite the small computer's relative "friendliness," efficient and efficacious use of the small computer is not effortless. To make the greatest and most effective use of the small computer, some study is required. The objective of this book is to present a fundamental introduction to the use of the small computer in the laboratory.

In preparing this book, a few assumptions are made abut the reader:

- Some familiarity with the scientific laboratory in the form of either professional practice or some study in an experimental science, such as a few lab courses in chemistry, physics, biology, or a related field
- An understanding of electricity such as that taught in high school physics
- Some experience in programming a computer [however, the language of that experience (FORTRAN, BASIC, Pascal, or even Logo) or the computer-type (mainframe or microcomputer) matters little. For the most part, examples are written in BASIC, even though the understanding of BASIC is not assumed.]

This book attempts to work from the top down. That is, the beginning chapters deal with the larger picture, defining the small computer and its place in the lab. Early in the book (Chapter 4), interfacing techniques are introduced at a level sufficient to satisfy many needs.

Ensuing chapters become somewhat more specialized. Chapters 5 and 6 present an operational approach to modern electronics, while Chapters 7 and 8 detail specialized transducers for sensing and controlling real-world phenomena. Communications, graphics, and computational methods, in Chapters 9 through 11, are presented with the laboratory computer in mind even though the subjects are not limited to lab applications, but are encountered by most small computer users. General topics related to the organization of a project round out Chapter 12.

A serious attempt has been made to limit the amount of jargon that must be learned as compared to what is needed for understanding other manuals and guides. A poster behind my word-processing computer shows Albert Einstein with the quotation "Everything should be made as simple as possible, but not simpler," and that has been a guide.

After studying this book, a scientific investigator should understand what types of problems a small lab computer can handle and how they are solved. Readers needing technical details concerning such topics as the advanced techniques of computer interface design, machine language programming, and numerical analysis are referred to advanced references.

Having gained the background provided here, the investigator should be able to use a small computer in the laboratory following one of two pathways. In many cases, simple interfacing tasks can be handled using only the information found in this text. On the other hand (and possibly of wider significance), a better understanding of laboratory computers should enable discussion of research directions involving computers with the electronics or data-processing personnel. The informed scientist will have the ability to communicate with them without being intimidated by strange concepts and unintelligible jargon.

Finally, the most desirable objective is that the investigator will come to treat the small computer as a useful, nonthreatening, and even enjoyable tool. With the barriers to the acceptance of computers dismantled, the computer can become a positive and challenging laboratory aid.

KENNETH L. RATZLAFF

Lawrence, Kansas
January 1987

Acknowledgments

The preparation of this book has been a most interesting challenge. There are many people who, knowingly or unknowingly, have made it possible. Expression of my appreciation is a pleasant task.

Christopher Gunn (Kansas University Graphics Lab) has been an expert editor, and the comments of Dr. Eugene Ratzlaff (IBM Instruments) and Professor John Walters (St. Olaf's College) have been invaluable.

Charlotte Pauls prepared most of the artwork, endeavoring to work within a schedule that was not her own. My colleague Tom Peters has been the source of new ideas and help; the Instrumentation Design Lab's steering committee has encouraged the effort.

Dr. Jean Johnson, Mathematics Department, University of Kansas, developed the tables in Appendix Two.

The most important are those who provided the nontechnical support. Jean, Francis, Audrey, Joe, Bob, Patrice, Jim, Carey, and Jean of the Lawrence Mennonite Fellowship have particularly helped to keep me on track. But most of all, my wife, Virginia, has given unfailing support, encouraging me and freeing me from shared family duties to spend time at the word processor, and Michael, Jonathan, and Rebekah (our children) have adjusted, especially during the final push. I express my sincere appreciation and gratitude to all of you.

K.L.R.

Contents

CHAPTER ONE

Introduction

1.1. INTRODUCTION

The reader of this book is likely to be engaged in science, as student or practitioner, with the knowledge that the small computer is having a profound impact on laboratory-based experimentation. For the scientist who would be inclined to read the book from the beginning, the place of the laboratory computer in the universe of computers is not necessarily obvious. That understanding can be useful in defining the scope of this study.

At the very outset, a *preliminary* definition of what will be considered to be a laboratory computer is necessary. We will consider it to be a machine that resides physically in or adjacent to the laboratory, has a limited number of users at any given time (usually one, but sometimes a few more), is programmable in one or more programming languages, and is capable of passing information to and from an experiment.

Furthermore, we want to learn to view the computer as "friendly"; that is, the computer should relate to its scientist users largely on their terms without the need for the user to undergo training for the status of "computer jockey." Indeed, over the past decade, the manufacturers of computers have changed the way that they view the computer user. People who use small computers are no longer expected to be computer professionals.

A parallel transformation also has occurred in the way society views the computer. This transformation is reflected in two important popular films, *2001, A Space Odyssey* and *Star Wars*. In the former, a single *highly centralized* computer, known as HAL, was capable of controlling many aspects of the lives and existence of the astronaut-protagonists until the computer finally was disabled; in HAL, we saw the artistic extension of the computer that instructed us not to "fold, spindle, or mutilate": that is, to work on the computer's terms. However, *Star Wars* depicted *distributed* computing power in the form of robots ("'droids") that were universally available to serve the user; the inherent capacity for good and evil remained with the humanoid users. (This view of robots in science fiction novels was spelled out in Isaac Asimov's *I, Robot* but did not become part of popular filmmaking until later.)

The equivalent transformation for the working scientist is equally profound. By bringing computers into the laboratory, the scientist in need of computing power is released from the whims and constraints of the "Computer Center"; control of the computer, to the extent that both the computing power and the split-second time response of the computer are directly available to the experiment, is brought into the laboratory.

1.1.1. Historical Development

One perspective on the laboratory computer's role can be gained by examining the historical development of computing—first in general, and then in the laboratory. Much like a tree (Fig. 1.1), this development has roots in several areas; these roots came together to form a single trunk as computers developed with fantastically increasing power. Although the quest for raw power continues, we will be interested in a fairly recent branch: a quest for the low-cost, easy-to-use power that has made routine laboratory operation feasible.

The first root in this historical tree was a calculating machine: specifically, the Analytical Engine of the Englishman Charles Babbage, designed in the 1830s. Babbage began with a small but clever device that he called the Difference Engine (Fig. 1.2), capable of automatically generating successive values of algebraic functions by the method of finite differences. The sequences were developed by Augusta Ada Lovelace Byron, daughter of poet Lord Byron, often called the first programmer. The machine was mechanical and only a small version was ever constructed successfully. However, that Difference Engine was successfully used for generating mathematical tables for several years.

Babbage, however, turned his attention to the more grandiose Analytical Engine which, although never actually constructed, was designed with the concepts of programmed sequences and punched card input and output (I/O). This made possible greatly increased precision through the sequential steps of multiple precision arithmetic and led to his claim, "I have converted the infinity of space, which was required by the conditions of the problem, into the infinity of time." The difficulty of machining the parts to the required precision led to the abandonment of the Analytical Engine.

A second root is often associated with a loom developed by J. M. Jacquard, patented in 1801. Punched cards were used to determine the sequence

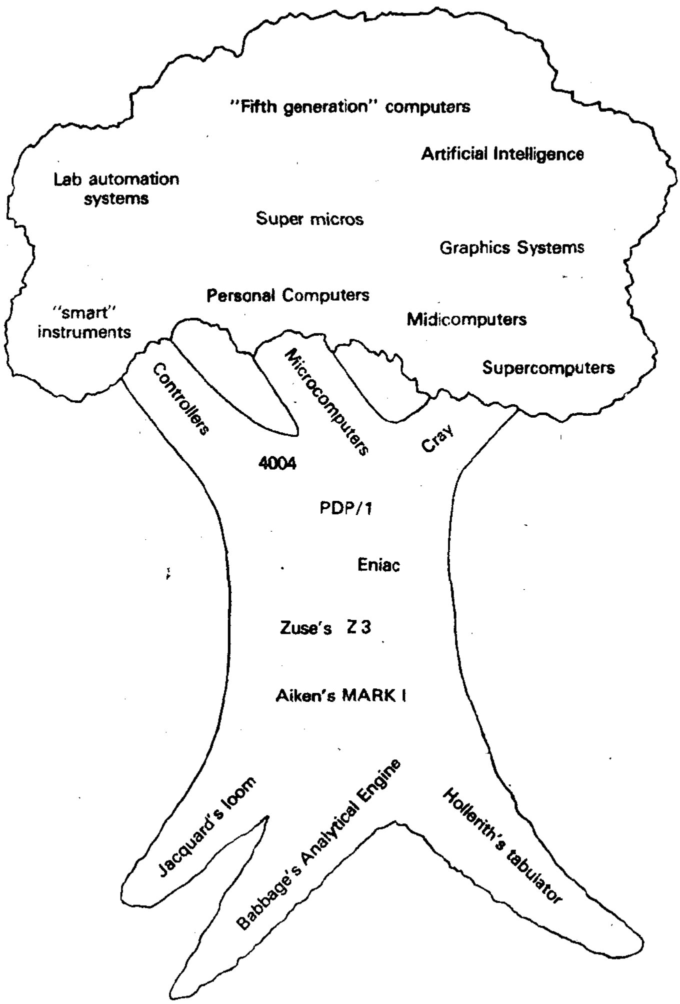

FIGURE 1.1 The historical family tree for computers. The roots lie in the nineteenth century, and the crown includes the present and the near future.

of operations, that is, to program the machine. This required logical decision making and is related to the concept of the servo-mechanism whereby a particular result (that is, the weaving operation) is compared with a control (the card) to determine if a change is necessary. The antitechnology Luddite rebellion developed partially in reaction to the introduction of this machine into England.

The name of Herman Hollerith is associated with a third root, the tabulating machine. The 1880 U.S. census showed that manual tabulation could not be completed within the census period, so Hollerith developed the

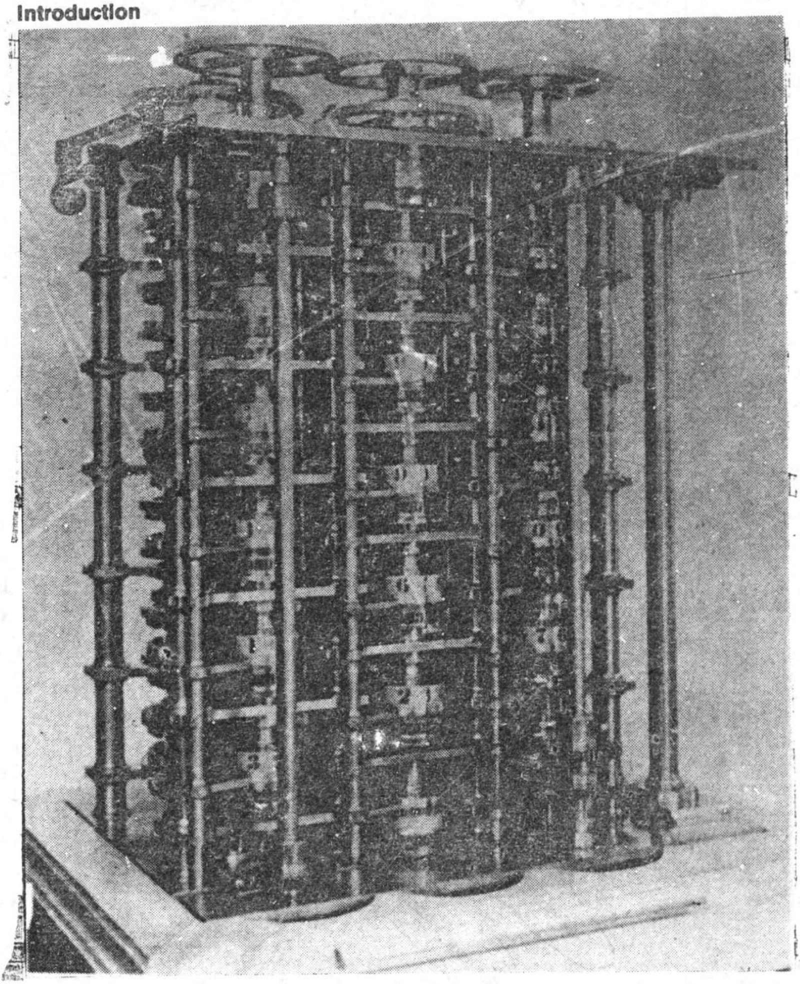

FIGURE 1.2 Part of Charles Babbage's first difference engine, 1832. [Courtesy of "Trustees of the Science Museum" (London).]

punched card machine for the 1890 census as a means of dramatically increasing the efficiency of sorting, tabulating, and storing large amounts of data. Even in the business end of computer development, Hollerith had a major impact since his Tabulating Machine Company became part of a company which led to the formation of International Business Machines (IBM) in 1924. His name is now applied to the field of a line of FORTRAN-language code which contains text.

By the late 1930s, the concept of a general-purpose digital computer was being developed. In Berlin, Konrad Zuse built the Z1 and Z2, storing data in memory constructed from old telephone relays. However, he broke new ground in computer logic with the Z3, which used floating-point representation to store 64 numbers, was program-controlled, and incorporated digital logic; it was the best working computer produced until 1941.

At about this time, Howard Aiken, a physics professor at Harvard, began a collaboration with four IBM engineers that resulted in the creation of the Mark I. It contained more than 3000 relays for data memory and weighed two tons! This machine was immediately put to work to support the U.S. Navy for classified computations.

The next step—to speed up operation limited by mechanical relays—was to introduce electronic computation. Zuse recognized this need but failed to get support from war-torn Germany, and Aiken distrusted electronics. In the absence of progress by these pioneers, the Electronic Numerical Integrator and Calculator (ENIAC) became the next major step. It was completed in 1946 at the University of Pennsylvania's Moore School of Engineering. The ENIAC led the way on a grandiose scale; over 19,000 vacuum tubes were incorporated in the device, consuming 200 KW of power. Although difficult to program, it was more than 3000 times faster in execution than the Mark 1.

The ENIAC, however, was based on the fundamental concepts embodied in the prototype computer of Dr. John V. Atanasoff at Iowa State College (now University), built in 1939. Atanasoff was the first to use the base-2 numbering system, dynamic electronic memory, and calculation by logic rather than enumeration.

For most of the next two decades, the emphasis in large computer development—and there were no other kinds—was on increased speed of operation and increased storage. The driving force for computers capable of handling scientific computations came from the military and aerospace sectors, but the business community also saw the value of computers for data management. Many companies emerged, spurred on by the demand of the consumer.

1.1.2. The Small Computer

In 1960, Digital Equipment Corporation (DEC) released the Programmed Data Processor 1 (PDP-1), the first of what would become known as the minicomputer. In its overall capability, it equalled the best computer of a decade earlier, but at a price of under $100,000.

Then, in 1965, the first computer built on an assembly line was produced, the DEC PDP-8. Tens of thousands of machines in this series were sold. The PDP-8 marked a significant shift in the philosophy of computers. Recall that a decade earlier computational capability was the overriding consideration. The cost and complexity of a computer was much too great to tie it directly to experiments or instruments in the real world, since the computer would then be required to operate in the time frame of actual experimental events (or in real-time, to introduce the jargon term). That would unacceptably require the computer to wait on the experiment or on other real-time events.

The cheaper, more rugged, and smaller minicomputer generated applications in process control, business, communications, and aerospace activities. It also found its way into scientific activities, primarily for the control of sophisticated experiments and instruments. The latter category accounted for little more than 10% of the market, however.

By 1970, there were scores of manufacturers of "minis" selling products in the $15,000 to $200,000 range. Clearly, the market could not support this proliferation of products, and a shakeout occurred, leaving a handful of dominant companies including recent leaders DEC, Data General, and Hewlett–Packard. However, the potential of such a machine for doing more than computing was clear to many scientists; it could take information directly from an experiment and use that information in the control of the experiment.

The development of large scale integration (LSI) of electronic components also was well underway. Powerful calculator circuits were beginning to drive the new and highly competitive hand-held calculator market. In 1968, a Japanese calculator company had approached Intel Corporation, a small electronics company, and asked them to design circuits for a scientific calculator. Engineer Ted Hoff, noting that the company PDP-8 computer's central processing board was less complex than the proposed calculator circuit, suggested that a fully *programmable* unit could do the job. The calculator company was not interested, but Intel management approved the project. The result, in 1971, was the Intel 4004, a 4-bit microprocessor with 2300 transistors, 45 instructions, and the capacity to address 4000 bytes. It quickly found its way into various controllers and timers, and was followed by a variety of new microprocessors with increasing internal complexity and greater capabilities.

By 1978, the Intel 8086 single-integrated circuit microprocessor had 29,000 transistors, 133 instructions, and could address a megabyte. Since then, microprocessor development has substantially surpassed the 8086 in complexity, addressability, and speed; with 32-bit microprocessors coming of age, no end is in sight. Intel, National Semiconductor, and Motorola have emerged as the leaders.

Although possibly not the most sophisticated microprocessor, the Intel 80386 chip is representative of the latest generation. By comparison, it has over 250,000 transistors on a single integrated circuit. The power of the instruction set, the operation speed, and the memory capacity have all been increased, and the need for other circuits to support the operation of the microprocessor has decreased (Fig. 1.3).

Hand-in-hand with microprocessor development came the production of integrated circuits that support the microprocessor: Inexpensive, compact memory circuits enable large capacity in a small unit, while integrated control chips for various peripheral devices (disk drives, graphic output systems, terminals, etc.) provide convenience and efficiency. The fully developed

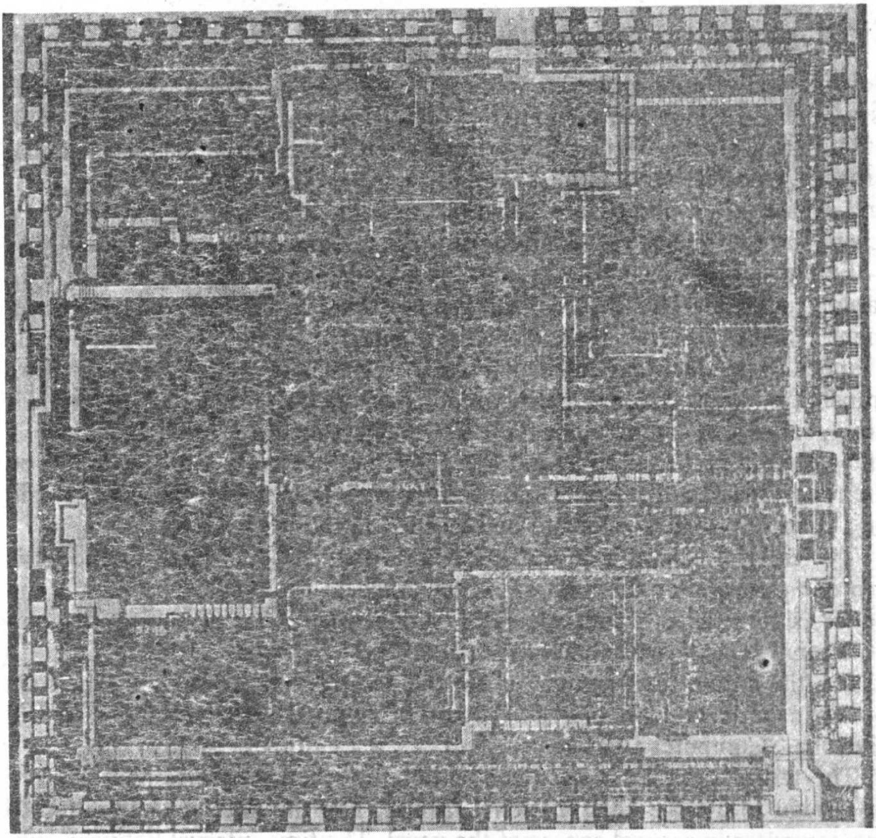

FIGURE 1.3. An Intel 80386 integrated circuit. Over 250,000 transistors are integrated into this silicon chip. (Courtesy Intel Corporation.)

microcomputer system has arrived in time for the challenges of the scientific laboratory.

New central processing units (CPUs) are being developed for data communications, calculations, data acquisition, signal processing, machine and instrument control, among other specialized applications. The CPUs that are used in personal/laboratory computers, however, are usually less specialized.

1.2. THE MODES OF COMPUTER OPERATION

The computer can be used to process the data and control the parameters of a scientific experiment in several ways. If we can avoid excessive rigidity at this stage, three modes of an experiment–computer relationship can be identified. As will become obvious, these modes are linked closely with the historical development described in the previous section.