COMPUTER TECHNOLOGY IN NEUROSCIENCE

Paul B. Brown

A Halsted Press Book

COMPUTER TECHNOLOGY IN NEUROSCIENCE

EDITED BY

PAUL B. BROWN

West Virginia University Medical Center
Morgantown

HEMISPHERE PUBLISHING CORPORATION
Washington London

A HALSTED PRESS BOOK

JOHN WILEY & SONS
New York London Sydney Toronto

Hemisphere Publishing Corporation
1025 Vermont Ave., N. W., Washington, D. C. 20005

Distributed solely by Halsted Press, a Division of John Wiley & Sons, Inc., New York.

1 2 3 4 5 6 7 8 9 0 PUPU 7 8 4 3 2 1 0 9 8 7 6

Library of Congress Cataloging in Publication Data

Main entry under title:

Computer technology in neuroscience.

1. Neurobiology–Data processing–Congresses.
I. Brown, Paul Burton, 1942–
QP357.5.C65 612'.8' 02854 76-6884
ISBN 0-470-15061-0

Printed in the United States of America

CONTRIBUTORS

R. Abba Department of Biological Sciences, Columbia University, New York, New York 10027

R. D. Benham Interactive Mini-Systems, Inc., Kennewick, Washington 99336

R. Bornholdt Department of Biological Sciences, Columbia University, New York, New York 10027

J. J. Capowski School of Medicine–Department of Physiology, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina 94305

T. O. Clarke Department of Neurology, Stanford University Medical Center, Stanford, California 94305

L. S. Davidow Department of Biology, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

M. L. Dierker Department of Anatomy, Washington University, St. Louis, Missouri 63130

C. E. Donaghey Department of Industrial Electrical Engineering, University of Houston, Houston, Texas 77004

K. Drake Poughkeepsie, New York 12603

D. J. Duffy Hospital Systems, West Virginia University Medical Center, Morgantown, West Virginia 26506

L. Dunkel Department of Physiology and Biophysics, University of Washington, Seattle, Washington 98195

D. H. Eldredge Central Institute for the Deaf, St. Louis, Missouri 63110

A. S. French Physiology Department, University of Alberta, Edmonton 7, Alberta, Canada

J. W. Froelich Department of Physiology and Biophysics, West Virginia University Medical Center, Morgantown, West Virginia 26506

J. D. Gerard Division of Biophysical Science, University of Alabama, Birmingham, Alabama 35294

G. W. Harding Department of Physiology and Biophysics, University of Washington, Seattle, Washington 98195

D. K. Hartline Biology Department, University of California at San Diego, La Jolla, California 92037

H. Howland Department of Neurobiology and Behavior, Cornell University, Ithaca, New York 14850

T. H. Kehl Department of Physiology and Biophysics, University of Washington, Seattle, Washington 98195

B. Kobler Biotechnology Program, Carnegie-Mellon University, Pittsburgh, Pennsylvania 15213

C. Levinthal Department of Biological Sciences, Columbia University, New York, New York 10027

E. R. Macagno Department of Biological Sciences, Columbia University, New York, New York 10027

J. Macy, Jr. Division of Biological Sciences, University of Alabama, Birmingham, Alabama 35294

J. Marler Department of Physiology and Biophysics, West Virginia University Medical Center, Morgantown, West Virginia 26506

T. W. McIntyre Department of Physiology and Biophysics, West Virginia University Medical Center, Morgantown, West Virginia 26506

R. McKown Department of Physiology and Biophysics, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

T. Medlin National Institute of Dental Research, Bethesda, Maryland 20014

S. L. Moise, Jr. Center for Health Sciences, University of California, Los Angeles, California 90021

H. Moraff New York Veterinary College Computer Facility, Cornell University, Ithaca, New York 14850

R. H. Olch Computer Science Department, University of California, Los Angeles, California 90024

D. O'Leary Department of Otolaryngology, University of Pittsburgh School of Medicine, Pittsburgh, Pennsylvania 15213

C. Parisot Brain Research Institute, University of California, Los Angeles, California 90024

J. H. Peacock Department of Neurology, Stanford University Medical Center, Stanford, California 94305

K. H. Pribram Psychology Department, Stanford University, Stanford, California 94305

B. J. Ransil Clinical Research Center, Beth Israel Hospital, Boston, Massachusetts 02215

W. S. Rhode Laboratory Computer Facility, University of Wisconsin, Madison, Wisconsin 53706

W. Roberts Department of Biology, University of California at San Diego, La Jolla, California 92037

C. Roby Department of Physiology and Biophysics, West Virginia University Medical Center, Morgantown, West Virginia 26506

D. A. Ronken Central Institute for the Deaf, St. Louis, Missouri 63110

A. C. Sanderson Biotechnology Program, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213

M. J. Shantz Information Science Department, California Institute of Technology, Pasadena, California 91125

V. Soni Laboratory Computer Facility, University of Wisconsin, Madison, Wisconsin 53706

L. Traini Department of Otolaryngology, University of Pittsburgh School of Medicine, Pittsburgh, Pennsylvania 15213

C. Tountas Department of Biological Sciences, Columbia University, New York, New York 10027

A. L. Towe Department of Physiology and Biophysics, University of Washington, Seattle, Washington 98195

J. Vidal Computer Science Department, University of California, Los Angeles, California 90024

C. Wall Department of Otolaryngology, University of Pittsburgh School of Medicine, Pittsburgh, Pennsylvania 15213

D. O. Walter Brain Research Institute, University of California, Los Angeles, California 90024

D. Walter Department of Physiology and Biophysics, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

D. F. Wann Department of Electrical Engineering, Washington University, St. Louis, Missouri 63130

M. P. White Division of Biological Sciences, University of Alabama, Birmingham, Alabama 35294

W. I. Wood Department of Biochemistry and Molecular Biology, Harvard University, Cambridge, Massachusetts 02183

PREFACE

This volume presents, insofar as possible, computer technology currently being applied to neuroscience research. The chapters represent independent development, in a number of laboratories, of workable computer systems to be applied. None of the systems presented can achieve all the objectives of all the laboratories, and most of the systems utilize a hardware environment that is more than the minimum needed. Each system required substantial investment of time and money for development (usually with little use made of similar systems developed elsewhere).

This work presents the first part of a two-part project aimed at reducing unnecessary inefficiencies involved in the independent implementation of computer-aided research from scratch in hundreds of laboratories around the country. Its objectives are: to specify a minimum hardware environment needed to perform each research task, with options to expand or adapt to existing hardware systems; to specify a minimum software package to perform each task or multi-task project; and to design these systems for easy modification and implementation. Part two will involve the implementation of these specifications by developing all the specified software. When this is accomplished, the package will be made available to the research community. Further expansions and refinements will be made available as need for them develops and as advances in computer science are made.

The need for this project is clear. To those who offer support for neuroscience research, duplication of effort in the development of computer research facilities is an unacceptable luxury. If the development and maintenance costs for software can be minimized or even eliminated, granting agencies may come to recognize well-developed general-purpose research computer systems as routine items of equipment. The development of inexpensive processors and peripherals, all of which can be assembled in the laboratory by technicians, should encourage such a trend. The possibility of a computer system that can be used by neuroscientists who have a minimum of computer science background should increase the use of such systems and open up new avenues of research not previously available to most neuroscientists.

This volume will alert researchers to the types of research made possible with computers applied to neurobiology. In some cases, the specific requirements for useful computer research systems are indicated. In others, specific software procedures and information are presented, which readers can apply in their own laboratories. Every effort was made to accurately represent the great breadth of computer applications in modern neurobiology.

I wish to thank the many neuroscientists from around the country who have

taken time from their research programs to share their expertise. Most of them are not computer scientists, but biologists, whose major activity is biological research.

This volume is an outgrowth of a symposium held at West Virginia University in April, 1975. It was the first of three workshops funded by The National Science Foundation (grant number DCR74-24765) for the specification of a general-purpose computer language to be used in neurobiology research. Preparation of the camera-ready copy was accomplished using the university's system 360 and the Department of Physiology and Biophysics' PDP-12. Cheryl Seim, Dr. T. W. McIntyre, and Darrell Duffy provided immeasurable assistance in making these systems work.

Paul B. Brown

CONTENTS

chapter 1

DESIGN AND DEVELOPMENT OF A MULTILABORATORY COMPUTER COMPLEX FOR BIOMEDICAL RESEARCH

J. MACY, JR. AND M. P. WHITE
Division of Biological Sciences
University of Alabama
Birmingham, Alabama

The Multiple Laboratory Computer Center (MLCC) is a multiple-processor, distributed logic system designed to support simultaneously the on-line computing needs of many independent laboratories. It was designed to handle a wide range of data-rate requirements from these laboratories, and to provide reasonable efficiency by shared use of central resources.

This system is a natural outgrowth of previous experience with on-line hybrid computing in neurophysiology, neurology, and physiology laboratories during the 1960's. (1) This experience was the basis for the development of methods for on-line and real-time analysis, and closed loop experimental control. It established the methods needed to handle data rates (sampling rates) required for these experiments. These rates ranged from 10 samples per second to more than 50,000 samples per second. Sampling rates were chosen to preserve the bandwidth and avoid aliasing, for complex waveforms, or to preserve a needed time resolution in the detection and characterization of specific neural events.

The work described in this chapter was supported in part by the U. S. Public Health Service, National Institute of Health Grant RR00145.

This experience led to the development of rather specific forms of "real-time" computing, and established the special needs of such computing. A brief discussion of the nature of "real-time" computing will point out these distinctions. When this term is used, it appears to mean many different things to different people. Experience with one form of on-line computing really gives very little idea of what any other kinds of real-time computing actually involve. Probably the simplest and most common form of real-time computing is used by an airline reservation system, or any other system which provides terminals into which a human can put information and can later make inquiries to get some part of the information back. The terminal may be a teletype or some combination of cathode ray screen and keyboard. This "real-time system" can be defined as a system in which the acceptable time for the computer performance is determined by the user and not by the program. The computer, in other words, must keep up with the demands of the real world. Most timeshared multiple-user computing systems have access times on the order of seconds, or some fraction of a second and, as the number of users and the usage of each terminal increases, the system reacts to the increased load by decreasing its performance. If the load is very heavy on a conventional time-shared system, there is a longer wait for response from the terminal. This is probably the only real-time system in which such a wait is acceptable. If a human at the terminal has to wait three seconds instead of one second he may become a bit annoyed but no data is lost or mangled because of the wait.

The next order of magnitude of speed for real-time systems is the world of the process-control system. These are the systems used in many factories, steel mills, and oil refineries to control machinery and equipment. This is, in effect, the kind of system and time scale also used in patient monitoring in the intensive care units and for delivering patient care directly by computer. The response times are generally in milliseconds. In monitoring an EKG, for such things as arrhythmias, for example, the time resolution needed is milliseconds.

The order of magnitude of real-time, on-line computing for which the MLCC was designed originated from the demands of neurophysiology laboratories, and depends on microsecond response and additional special features.

Some examples from neurophysiological work will illustrate this. At the time the system was first conceived, much of our work was based on multiple-electrode, single-cell measurements, because we were particularly interested in the coding schemes of different areas of the nervous system and in the time series performance of some of these cells. We were attempting to make measurements for which it was necessary to convert the data continuously amd accumulate statistics over periods of many minutes. We had runs which would go continuously for as long as forty-five minutes, and we were generally working with two to five channels. To get the time resolution required for most of the measurements implied a sampling rate for each channel of at least 5,000 samples per second, sometimes as many as 10,000 or 20,000. The time between samples for each channel was less than 200 microseconds, sometimes as little as 20 microseconds. This is very different from the situation in an inquiry terminal or in intensive care monitoring. "Real real-time" could be considered as a context in which the time between the arrival of samples is not more than eight or ten times the basic machine cycle of the computer being used. The computer is no longer very much faster than the real world with which it is trying to cope but operates at about the same speed as the world being computed. Much of our work in the 1960's used a computer with a basic machine cycle of 6.5 microseconds, and an average execution time for commands of 15 microseconds. The interval between samples in this research was between 50 and 200 microseconds. In the worst case there was time between samples to execute three instructions. Furthermore, programming to take in 50, 100 or 5,000 samples and store them to be computed later was not possible, because in many cases it was necessary to keep up on a continuous basis wih a continuous process. Data points came in at a rate of 10,000 to 50,000 a second for about half an hour. The only possible technique was to compute fast enough to keep up, starting to build the answer and throwing away as much as possible of the raw samples. If the samples arrive on many channels at such a speed there must be some provision to dispose of irrelevant or combinable data, because no machine has enough storage to bring in and store such a flood.

For some processes it is possible to compress the data. "Outboard" analog or hybrid hardware will recognize in one way or another the particular phenomenon of interest and start sampling when it occurs. This happens frequently in monitoring where there is no real interest in analyzing every heartbeat for a twenty-four hour period but only in checking and accumulating derived measurements at stated intervals. Intermittently a series of heartbeats will be sampled, the complete analysis performed and the raw data discarded.

The MLCC was designed to meet the requirements of laboratories with a legitimate need for sampling rates up to 50,000 samples per second per channel. Some laboratories have requirements well below 1,000 samples per second per channel, so a range of time scales is needed to match the full range of requirements.

In many cases, there is an additional requirement for local control of the experiment in response to the already acquired data so that the progress of the experiment is actually dependent upon the results just processed: a closed-loop situation.

We started our multilaboratory design on the assumption that no computer can keep up with the flow of all the data from ten or more laboratories with 50 kc sampling rates per channel. It would be necessary to install additional preprocessing logic at the laboratory to cut down the actual rate of arrival of data at the central computer. In such a system, the logic would be spread and distributed so that the computing power with high precision, floating point, versatile commands and other refinements would be located at the center of the system. In general, the operations to be performed on the samples as they first come in, at the initial high rate of speed, are fairly simple and the conversions will be, at most, twelve bit conversions, so that extremely high precision is not essential. When the preprocessing operations have been performed at a very high rate of speed, the resulting sampling rate may be one-tenth of the original rate (perhaps 100 per second) and the machine can operate as a more conventional time-sharing system with the effectively reduced time scale. The burden of continuously computing and keeping up has been shifted out so that a small amount of relatively inexpensive logic can be dedicated to preprocessing the data full-time, unshared by other users.

The design philosophy of our system is based on four considerations. The first is the necessity for the tight control of time. In terms of Central Processing Unit