Advanced Microprocessors

Edited by
Amar Gupta
Hoo-min D. Toong
Sloan School of Management
Massachusetts Institute of Technology

Advanced Microprocessors

Edited by
Amar Gupta
Hoo-min D. Toong
Sloan School of Management
Massachusetts Institute of Technology

A volume in the IEEE PRESS Selected Reprint Series,
prepared under the sponsorship of the
IEEE Computer Society.

IEEE
PRESS

The Institute of Electrical and Electronics Engineers, Inc., New York

1884 · 1984
A CENTURY OF ELECTRICAL PROGRESS

PRINTED IN THE UNITED STATES OF AMERICA

Sole Worldwide Distributor (Exclusive of the IEEE)

JOHN WILEY & SONS, INC.
605 Third Ave.
New York, NY 10158

Wiley Order Number: 471-88176-7

IEEE Order Number: PC01602

Library of Congress Cataloging in Publication Data
Main entry under title:

Advanced microprocessors.

(IEEE Press selected reprint series)
"Prepared under the sponsorship of the IEEE Computer Society."
Includes bibliographies and indexes.
1. Microprocessors—Addresses, essays, lectures.
I. Gupta, Amar. II. Toong, Hoo-min D. III. IEEE Computer Society.
QA76.24.A34 1983 001.64 83-6092
ISBN 0-87942-167-3

Advanced Microprocessors

OTHER IEEE PRESS BOOKS

Biological Effects of Electromagnetic Radiation, *Edited by John M. Osepchuk*
Engineering Contributions to Biophysical Electrocardiography, *Edited by Theo C. Pilkington and Robert Plonsey*
The World of Large Scale Systems, *Edited by James D. Palmer and Richard Saeks*
Electronic Switching: Digital Central Systems of the World, *Edited by Amos E. Joel, Jr.*
A Guide for Writing Better Technical Papers, *Edited by C. Harkins and D. L. Plung*
Low-Noise Microwave Transistors and Amplifiers, *Edited by H. Fukui*
Digital MOS Integrated Circuits, *Edited by M. I. Elmasry*
Geometric Theory of Diffraction, *Edited by R. C. Hansen*
Modern Active Filter Design, *Edited by R. Schaumann, M. A. Soderstrand, and K. R. Laker*
Adjustable Speed AC Drive Systems, *Edited by B. K. Bose*
Optical Fiber Technology, II, *Edited by C. K. Kao*
Protective Relaying for Power Systems, *Edited by S. H. Horowitz*
Analog MOS Integrated Circuits, *Edited by P. R. Gray, D. A. Hodges, and R. W. Brodersen*
Interference Analysis of Communication Systems, *Edited by P. Stavroulakis*
Integrated Injection Logic, *Edited by J. E. Smith*
Sensory Aids for the Hearing Impaired, *Edited by H. Levitt, J. M. Pickett, and R. A. Houde*
Data Conversion Integrated Circuits, *Edited by Daniel J. Dooley*
Semiconductor Injection Lasers, *Edited by J. K. Butler*
Satellite Communications, *Edited by H. L. Van Trees*
Frequency-Reponse Methods in Control Systems, *Edited by A.G.J. MacFarlane*
Programs for Digital Signal Processing, *Edited by the Digital Signal Processing Committee, IEEE*
Automatic Speech & Speaker Recognition, *Edited by N. R. Dixon and T. B. Martin*
Speech Analysis, *Edited by R. W. Schafer and J. D. Markel*
The Engineer in Transition to Management, *I. Gray*
Multidimensional Systems: Theory & Applications, *Edited by N. K. Bose*
Analog Integrated Circuits, *Edited by A. B. Grebene*
Integrated-Circuit Operational Amplifiers, *Edited by R. G. Meyer*
Modern Spectrum Analysis, *Edited by D. G. Childers*
Digital Image Processing for Remote Sensing, *Edited by R. Bernstein*
Reflector Antennas, *Edited by A. W. Love*
Phase-Locked Loops & Their Application, *Edited by W. C. Lindsey and M. K. Simon*
Digital Signal Computers and Processors, *Edited by A. C. Salazar*
Systems Engineering: Methodology and Applications, *Edited by A. P. Sage*
Modern Crystal and Mechanical Filters, *Edited by D. F. Sheahan and R. A. Johnson*
Electrical Noise: Fundamentals and Sources, *Edited by M. S. Gupta*
Computer Methods in Image Analysis, *Edited by J. K. Aggarwal, R. O. Duda, and A. Rosenfeld*
Microprocessors: Fundamentals and Applications, *Edited by W. C. Lin*
Machine Recognition of Patterns, *Edited by A. K. Agrawala*
Turning Points in American Electrical History, *Edited by J. E. Brittain*
Charge-Coupled Devices: Technology and Applications, *Edited by R. Melen and D. Buss*
Spread Spectrum Techniques, *Edited by R. C. Dixon*
Electronic Switching: Central Office Systems of the World, *Edited by A. E. Joel, Jr.*
Electromagnetic Horn Antennas, *Edited by A. W. Love*
Waveform Quantization and Coding, *Edited by N. S. Jayant*
Communication Satellite Systems: An Overview of the Technology, *Edited by R. G. Gould and Y. F. Lum*
Literature Survey of Communication Satellite Systems and Technology, *Edited by J.H.W. Unger*
Solar Cells, *Edited by C. E. Backus*
Computer Networking, *Edited by R. P. Blanc and I. W. Cotton*
Communications Channels: Characterization and Behavior, *Edited by B. Goldberg*
Large-Scale Networks: Theory and Design, *Edited by F. T. Boesch*
Optical Fiber Technology, *Edited by D. Gloge*
Selected Papers in Digital Signal Processing, II, *Edited by the Digital Signal Processing Committee, IEEE*
A Guide for Better Technical Presentations, *Edited by R. M. Woelfle*
Career Management: A Guide to Combating Obsolescence, *Edited by H. G. Kaufman*
Energy and Man: Technical and Social Aspects of Energy, *Edited by M. G. Morgan*

Contents

Acknowledgment

This book has been made possible by the encouragement provided by the IEEE Press Editorial Board. We are also grateful to the staff of the IEEE Press for bringing out this book in minimum time. Last, but not least, we thank Nona Jenkins for her valuable assistance in putting this book together.

AMAR GUPTA
HOO-MIN D. TOONG

Part I Overview

THE TERM "microprocessor" was first used in 1972. However, the era of microprocessors commenced in 1971 with the Intel 4004, a "microprogrammable computer on a chip" composed of an "integrated CPU complete with a four-bit parallel adder, 16 four-bit registers, an accumulator and a push-down stack on a chip" [1]. The 4-bit 4004 CPU contained 2300 transistors, and could execute 45 different instructions. Subsequently, 8-bit, 16-bit, and 32-bit microprocessors were introduced in 1972, 1974, and 1981, respectively. Today, the Hewlett-Packard 32-bit microprocessor contains 450 000 transistors and offers a repertoire of 230 instructions. In less than twelve years, we have seen four generations of microprocessors. For a long time, the computer revolution was considered unparalleled in history in terms of its pace. The progress in the domain of microprocessors is even more significant, and the pace two to three times faster than in the case of computers.

A microprocessor is the central arithmetic and logic unit of a computer, together with its associated circuitry, scaled down so that it fits on a single silicon chip (sometimes several chips), holding tens of thousands of transistors, resistors, and similar circuit elements [2]. Microprocessors are characterized along several dimensions as follows:

Chip Technology

In view of the large number of transistors, most manufacturers have opted to fabricate mircroprocessors using MOS (metal oxide semiconductor) technology in preference to bipolar transistor technology. Currently, the most popular MOS technology is n-channel MOS (NMOS), by virtue of its high packing density and fast switching speeds. CMOS (complementary MOS) circuits provide faster speed and lower power consumption than circuits implemented with traditional PMOS and NMOS technology; the disadvantage of CMOS lies in its lower packing density. In coming years, CMOS may become the most popular technology for fabricating microprocessors [3].

Word Size

Word size reflects the basic unit of work for the microprocessor. A larger word size implies more processing power and addressing capabilities. In the early years of microprocessors, size of registers, size of internal instruction paths and data paths, and size of external instruction and data paths were all identical. This is rarely true now. Large external data paths require the chip package to have a high number of pins, which implies high packaging and production costs. Thus, chips nowadays tend to have larger internal paths than external paths. For example, the Motorola 68000 and the National NS16032 have 32-bit internal paths and 16-bit external paths. A true 32-bit microprocessor has all external paths and all internal units designed to communicate or process at least 32 bits in parallel.

Type of Microprocessor

Some microprocessors process all bits of one word in parallel. Others work with "slices" of data and/or instruction words. In the latter case, called "bit-slice architecture," several identical chips can be used to process different slices in parallel. In the past, the lack of ability to fabricate a large number of transistors on the same chip made it essential to use multiple bit-slice chips to obtain large word widths. Today bit-slice architecture offers potential for creating a customized CPU with a word length that is an integral multiple of the bit-slice width. However, for most applications, word sizes of 16 bits or 32 bits are more than adequate, permitting use of standard (nonbit-slice) 16-bit and 32-bit microprocessors.

Microprogramming

Early microprocessors used "hard-wired architectures" with functions determined by fixed circuit paths. A microprogrammed CPU, although inherently slower than a hard-wired CPU, offers greater flexibility in terms of easier incorporation of changes or additions to the instruction set. The trend is towards microprogrammed microprocessors. Only a few of these chips can be microprogrammed by the user.

Clock Frequency (Hertz)

This is the number of clock cycles per second of the fundamental driving clock circuit. Two clocks of the same frequency, but phase shifted with respect to each other, can be used to generate a clock of higher frequency. An increase in the clock frequency results in a proportionate decrease in the execution of an instruction.

Number of Instructions

The enhanced ability to fabricate large numbers of transistors on a chip facilitates implementation of a larger instruction set. These instructions are frequently of varying sizes, depending on instruction type, size of data, and addressing mode used. As the instruction size has increased, the complexity of the chip has increased, and so has the design effort, from under a man-year to over 100 man-years of engineering time [1]. To reduce this massive effort, reduced instruction set architectures have been implemented [4], [5]. In coming years, commercial microprocessors will not offer much larger instruction sets—individual instructions will, however, become more powerful.

Addressing Capability

Early microprocessors could reference only limited memory space. Larger word sizes enable direct addressing of larger memory space. In addition, a number of auxiliary addressing modes (such as indirect, indexing, autodecrementing) have become popular on microprocessors. Specialized memory management chips provide even more enhanced capabilities for efficient memory management. Finally, virtual memory facilities are becoming a standard feature on the newer microprocessors.

Number of Registers

Registers are required for arithmetic operations, for stack operations, for storing base and index values, and for a variety of other operations, depending on the architecture. General-purpose registers can be used for multiple uses. Some microprocessors offer general-purpose registers only, others dedicated only, but most offer some combination of the two.

Data Types

All microprocessors support data in the form of bytes and words. However, only some support data in the form of bits, binary coded decimal words, floating-point numbers, words longer than 4 bytes, and character strings. Floating-point capabilities are useful for scientific work. Character string manipulation capability is required for text editing applications. Auxiliary chips, co-processors, or slave processors are sometimes used to perform these functions. As technology improves, it will be feasible to incorporate more functions on the main chip itself.

Direct Memory Access (DMA) Capability

DMA capability enables a processor to offer a higher overall performance by allowing input and/or output to proceed concurrently with processing. In most cases, an auxiliary chip is used to take over the task of controlling input and output operations, leaving the microprocessor free to process instructions.

Software

Two decades back, the Burroughs 5000 series introduced a trend of architectures designed to support high-level languages alone. Today there is the Intel iAPX 432, designed to be programmed entirely in a high-level language; its system architecture is consciously oriented toward supporting ADA. This trend will become widespread in the industry as it enables reduced user programming costs. Some new microprocessors provide high-level language-oriented instruction sets and enhanced support for switching from one process to another.

Multiprocessing Capabilities

In order to increase computational bandwidth and/or system resilience, it becomes necessary to integrate several microprocessors in a single system. The overall throughput and efficiency of such systems is directly dependent on the hardware and software interconnection mechanisms supported by the basic microprocessor chips. Although all chips offer some facilities for multiprocessing, it is essential to examine exact features to determine overall maximum efficiency of multiprocessor configurations and to estimate software overheads.

Number of Chips

The Intel iAPX 432 comes as a three-chip set. Other 32-bit microprocessors require auxiliary chips to perform meaningful functions. Thus, high performance configurations are comprised of multiple chips. Single-chip microcomputers, on the other hand, contain processor, memory, and input/output logic on the same chip. Because of the chip area needed for functions other than processing, these single-chip microcomputers are less powerful than microprocessors fabricated using the same technology. Two conflicting trends will continue; that is, single-chip microcomputers will become increasingly complex and powerful, and multichip microprocessor-based configurations will be used in increasing numbers to undertake more complex tasks (database, transaction processing) that have traditionally been done on larger mainframes.

In addition to the above, cost, availability of support chips, second-sourcing considerations, reliability of product, upward software compatibility with an earlier chip, and the nature of applications will influence the choice of a microprocessor.

A comprehensive analysis of all the diverse issues indicated above is a massive task because of the incredible pace of the microprocessor revolution. Handbooks published by chip manufacturers highlight the merits of their products, but present meager information on the design problems and the faults of their products. Also, designers find it virtually impossible to meaningfully compare and contrast products of different manufacturers.

This reprint book is intended to fill this void. It is organized into six parts as follows.

Part I: Overview

An introduction to the realm of microprocessors, including a history of the development of microprocessors.

Part II: 16-bit Microprocessors

A comprehensive collection of articles on contemporary 16-bit microprocessors and auxiliary support chips. Almost all of these papers are written by the persons who designed the chips.

Part III: 32-bit Microprocessors

A description of several high performance 32-bit microprocessors, including chips designed exclusively for internal use within sponsoring organizations. These internal chips, harbingers of similar capability public domain products, present an interesting overview of emerging trends in microprocessors.

Part IV: Performance Comparisons

Comparison of performance of 16-bit and 32-bit microprocessors for several different application scenarios, using theoretical methods and benchmark programs.

Part V: Related Technologies

A state-of-the-art overview of two related technologies—bit-slice architecture and single-chip microcomputers.

Part VI: System Issues

Summary of standards for interconnection mechanisms and for languages.

Read in sequence, these parts provide a tutorial to the rapidly growing microprocessor field. Persons with previous background will find it convenient to refer directly to the part that they are interested in. In proportion to the greater interest and higher usage of the newer chips, this book concentrates on 16-bit and 32-bit microprocessors and contains papers written during the 1980's only. The Bibliography refers to papers published in the 1980's, and provides sources of additional information on related topics.

This part (Part I) includes three general papers that look at the field of microprocessors from three different perspectives. In the first article, Dennis Moralee provides a history of important landmarks during the first decade of microprocessors and an analysis of how the architectural evolution of microprocessors is linked to that of mainframe computers and minicomputers. The second article, by Ian H. Witten, examines the economics of information processing, coming to a significant conclusion—as hardware costs plummet, it becomes increasingly relevant to decrease software costs by offering high-level languages; this article also shows the advantages of 16-bit processors over 8-bit processors in terms of extended addressing range, memory segmentation and protection, regularity of instruction set, string manipulation capabilities, and support of more data types. The third article, by Paul M. Russo, emphasizes the intimate relationship between trends in very-large-scale-integration (VLSI) and the evolution, usage, and system design of microprocessors. The paper shows that the advent of early microprocessors caused the LSI development to be re-oriented towards design of chips with potential for use in industrial, commercial, and consumer applications.

Before the turn of this century, the population of microprocessors in use will exceed the population of people living on this planet. Hopefully, even in light of being outstripped in terms of numbers, this book will help the human race to remain in control!

References

[1] R. N. Noyce and M. E. Hoff, Jr., "A history of microprocessor development at Intel," *IEEE Micro*, vol. 1, pp. 8–21, Feb. 1981.

[2] H. D. Toong, "Mircroprocessors," *Sci. Amer.*, pp. 146–161, Sept. 1977.

[3] D. L. Wollesen, "CMOS LSI—The computer component process of the 80's," *Computer*, pp. 59–67, Feb. 1980.

[4] D. A. Patterson and C. H. Sequin, "A VLSI RISC," *Computer*, vol. 15, pp. 8–21, Sept. 1982.

[5] G. Radin, "The 801 minicomputer," in *Proc. Symp. Architectural Support for Programming Languages and Operating Syst.*, Mar. 1–3, 1982, pp. 39–47.

Bibliography

[1] "Microprocessors," *Elec. Eng.*, pp. 60–85, Sept. 1981.

[2] R. Bernhard, "More hardware means less software," *IEEE Spectrum*, pp. 30–37, Dec. 1981.

[3] R. M. Cushman, "CMOS microprocessor and microcomputer ICs," *EDN*, pp. 88–100, Sept. 29, 1982.

[4] H.M.J.M. Dortmans, "Application of microprocessors," *J. Phys. E. Sci. Instrum.*, vol. 14, pp. 777–782, 1982.

[5] D. A. Fairclough, "A unique microprocessor instruction set," *IEEE Micro*, vol. 2, pp. 8–17, May 1982.

[6] E. H. Frank and R. F. Sproull, "Testing and debugging custom integrated circuits," *ACM Comput. Surveys*, vol. 13, pp. 425–451, Dec. 1981.

[7] H. W. Lawson, Jr., "New directions for micro- and system architectures in the 1980s," in *Proc. Nat. Comput. Conf.*, 1981, pp. 57–62.

[8] C. M. Lee and C. G. Lin-Hendel, "Current status and future projection of CMOS technology," in *Proc. Compcon Fall*, Sept. 20–23, 1982, pp. 716–719.

[9] S.-M. S. Liu, C.-H. Fu, G. E. Atwood, H. Dun, J. Langston, E. Hazani, E. Y. So, S. Sachdev, and K. Fuchs, "HMOS III technology," *IEEE J. Solid-State Circuits*, vol. SC-17, pp. 810–815, Oct. 1982.

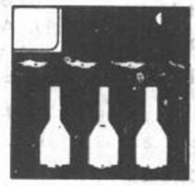
[10] D. J. McGreivy and K. A. Pickar, *VLSI Technologies Through the 80s and Beyond* (IEEE Computer Society Reprint Book). New York: IEEE Press, 1982.

[11] S. P. Morse *et al.*, "Intel microprocessors—8008 to 8086," *Computer*, pp. 42–60, Oct. 1980.

[12] D. A. Patterson and R. S. Piepho, "Assessing RISCs in high-level language support," *IEEE Micro*, pp. 9–19, Nov. 1982.

[13] R. Rice, *VLSI Support Technologies—A Tutorial* (IEEE Computer Society Reprint Book). New York: IEEE Press, 1982.

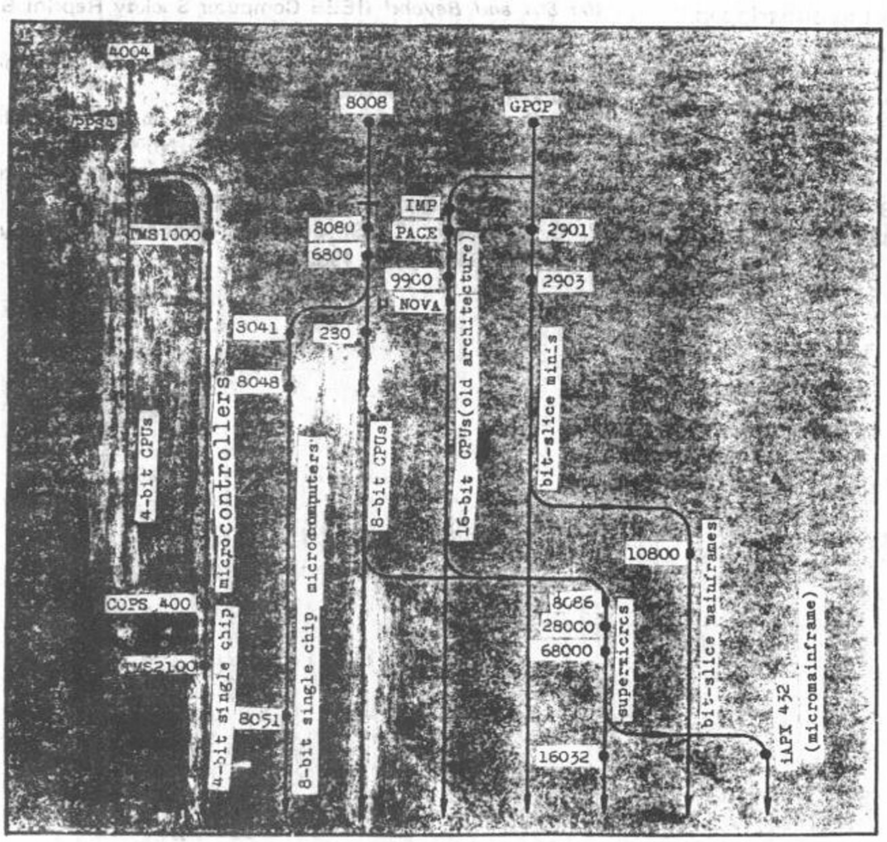
[14] P. C. Treleaven, "VLSI processor architectures," *Computer*, vol. 15, pp. 33–45, June 1982.



Microprocessor architectures: ten years of development

Ten years ago the first true microprocessor became commercially available. What were the origins of the microprocessor, how has it evolved since then, and how has its architectural evolution been linked to that of mainframe computers and minicomputers?

by Dennis Moralee



1 Genealogy of the microprocessor family

Just ten years ago, a small and at that time relatively little known semiconductor company called Intel launched an unusual new product, the MCS-4 'microprogrammed minicomputer' system based on a single-chip central processing unit (CPU) designated the 4004. This small PMOS device, containing the now very modest total of 2250 transistors, was in fact the first microprocessor, the very first of a now long line of microelectronics devices that, in only 10 years, have come to dominate modern electronics design and become the focus not only of intense engineering interest but also of an increasingly widespread public debate. Although at the time of its introduction the 4004 may have seemed to many to be little more than a minor technological curiosity unlikely to have any real long-term significance, with hindsight its development can now be seen as signalling the start of one of the most remarkable periods of technological change to occur in modern times. As the 4004's designers correctly predicted, the introduction of the new device was merely the first step along a whole new line of technological development: 'The MCS-4', they wrote,[1] 'is really only a beginning'.

However, even the 4004's designers would have had difficulty in correctly

Reprinted with permission from *Electron. and Power*, vol. 27, pp. 214–221, Mar. 1981.

predicting just how rapidly the microprocessor would evolve during its first 10 years. Some idea of how dramatic this evolution has been can be obtained by comparing the 4004 with the microprocessor device Intel launched just a few weeks ago, the iAPX 432 'micro-mainframe', an undoubted — if not directly lineal — descendant of the original MCS-4 family. Perhaps the most obvious comparison is in terms of 'raw' processing power: the 4004 was very much less powerful than even the basic minicomputers of the day, whereas the iAPX 432 is in some configurations as powerful as a contemporary midrange mainframe, the type of conventional computer that will typically fulfil all the traditional computing needs of a medium-sized manufacturing company.

In many ways more significant, however, is the comparison in terms of architecture, the distinctive functional organisation of the devices' computational resources. While the 4004, viewed as a general-purpose computing device, had an architecture that was primitive even when compared to the minicomputers of the day, the architecture of the iAPX 432 is not just more sophisticated than those of today's commonly used mainframes, it is in many respects more sophisticated than practically any general-purpose computing device yet put on the market. After only a single decade, in fact, microprocessor design has evolved from a situation in which it lagged far behind conventional-computer design to a place where it is beginning to take the lead.

Dramatically as such comparisons illustrate the rapid evolution of microprocessor design, however, they falsely suggest that this evolution has followed a single line of development. In fact, several very different types of microprocessor have evolved over the last 10 years, of which the general-purpose CPUs such as the iAPX 432 are perhaps the best known, if not necessarily the most important.

Microcontrollers

Still the most commonly used type of microprocessor device is the 4-bit 'single chip' *microcontroller*, a type that has changed only in detail from the first-generation chip-set such as the MCS-4, most importantly by the integration of the originally separate CPU, memory and I/O units onto a single chip.

Alone of all the microprocessor devices in production today, these microcontrollers still use the original PMOS technology, largely because of its extremely low cost, which tends to be an important factor in the low-end 'logic replacement' applications in which they are mainly used[2] (see Collie pp.236-239, this issue). More recently, these devices have been joined by the more powerful 'single-chip' *microcomputers*, essentially a reworking of the same concept but with a second-generation 8-bit CPU on the chip instead of a first-generation 4-bit unit: these devices, available in both NMOS and CMOS implementations, are typically used in the more demanding of the logic-replacement applications.

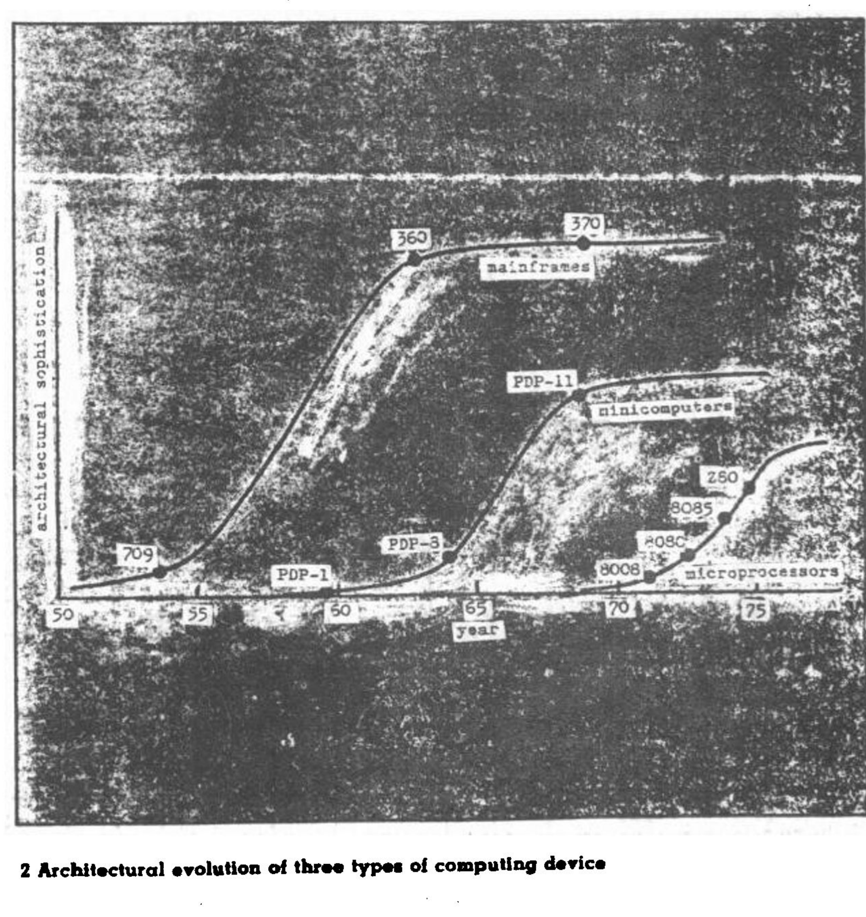
Bit-slice devices

At the other end of the processing-power scale, are the *bit-slice* devices, LSI processor components that are traditionally classed as a type of microprocessor, even though they differ in very many ways from all the other types. In use, several of these devices are combined to form the nucleus of a specific form of CPU, a *microprogrammed* processor that acts as a sort of 'computer within a computer', performing the overall CPU function not by the operation of a mass of hard-wired circuitry but by the execution of specialised *microprograms*. Working with these devices involves the use in the detailed internal design of the CPU, an advantage in certain specialised applications (see Clements pp.230-235, this issue), but an unjustified extra complication in applications for which 'ready-made' CPUs of the right characteristics are available. Because of this, bit slices have tended to be used in conventional microprocessor applications only when their intrinsically higher processing power has been required: the fast bipolar logic normally used in their construction has allowed the construction of CPUs in the mainframe performance category for some time.

Although the bit slices are thus a rather specialised form of microprocessor device, their ancestry can be traced back to the early days of microprocessor design, if not to the Intel 4004 at least to the National Semiconductor GPCP, a pioneering device that was developed almost simultaneously with the 4004, appearing only about a year afterwards. Since then, the bit slices have evolved from 2-bit and 4-bit models of limited throughput, via the now industry-standard 4-bit Advanced Micro Devices' 2901 and 2903, to newer 8-bit and even 16-bit models, the last mentioned being a complete 'unsliced' microprogrammable CPU.

Between these two very different types of somewhat specialised microprocessor device are perhaps the best-known microprocessors, the general-purpose single-chip CPUs. Because of their general-purpose nature, these devices have been used in the widest variety of microprocessor applications, ranging from low-end 'logic replacement' applications beyond the power of the microcontrollers and microcomputers, to high-end 'information system' applications once the province of conventional minicomputers. Originally descended from the 8-bit PMOS 8008, a specifically general-purpose device designed almost simultaneously with the 4004 and released only a few months later, this type of microprocessor has gone through three distinct generations, and with the introduction of the iAPX 432 seems currently about to enter a fourth.

After the introduction in 1974 of the first generation of these devices, typified by the 8008 itself, perhaps the greatest advance in their design was made with the adoption of the superior NMOS technology in 1974. The change



2 Architectural evolution of three types of computing device

to NMOS allowed more subsidiary functions to be incorporated on the chip, the architecture of the devices to be improved and their performance to be increased dramatically: the devices of this NMOS second generation, such as the 8080, 6800, 8085, Z80, 6502 etc., continue to be the best known microprocessors in use today. The fact that these influential mainstream devices all have an 8-bit architecture has led this wordlength to be associated almost exclusively with this generation of devices, but in fact a number of 16-bit devices date from this time: PACE, INS8900, 9900, microNOVA, CP1600 etc. The fact that these devices have a 16-bit wordlength does not, however, indicate *a priori* that they are more powerful or more sophisticated than their 8-bit contemporaries.

Third-generation CPUs

A range of more powerful and more sophisticated 16-bit devices did become available, however, with the introduction of the third generation CPUs in 1978. Based on the then newly developed high-density versions of NMOS, variously called HMOS, XMOS etc., this new generation of devices brought general-purpose microprocessors into the performance class of traditional minicomputers for the first time. Much more than 'souped up' versions of the second-generation devices, however, these new 'supermicros' are more importantly distinguished by their advanced architectures which, in some cases, outdo in general sophistication even those of the best-known mainframes.

The advanced architectures of these 'supermicros' — the Intel 8086 and its follow-up devices, the Zilog Z8000, the Motorola 68000 and the new National Semiconductor 16000 range — have not come about by accident, but by careful consideration by the microprocessor manufacturers of the characteristics that a high-end general-purpose microprocessor will have to possess in order to be a commercial success over the next decade or so.

After long development projects, which in some cases began in the early 1970s even before the second-generation NMOS devices were released, the manufacturers have all come out in favour of certain architectural concepts that are now reflected, with differing degrees of emphasis, in all the third generation designs. The current third-generation of general-purpose microprocessors could, in fact, be quite accurately called the 'advanced architecture' generation.

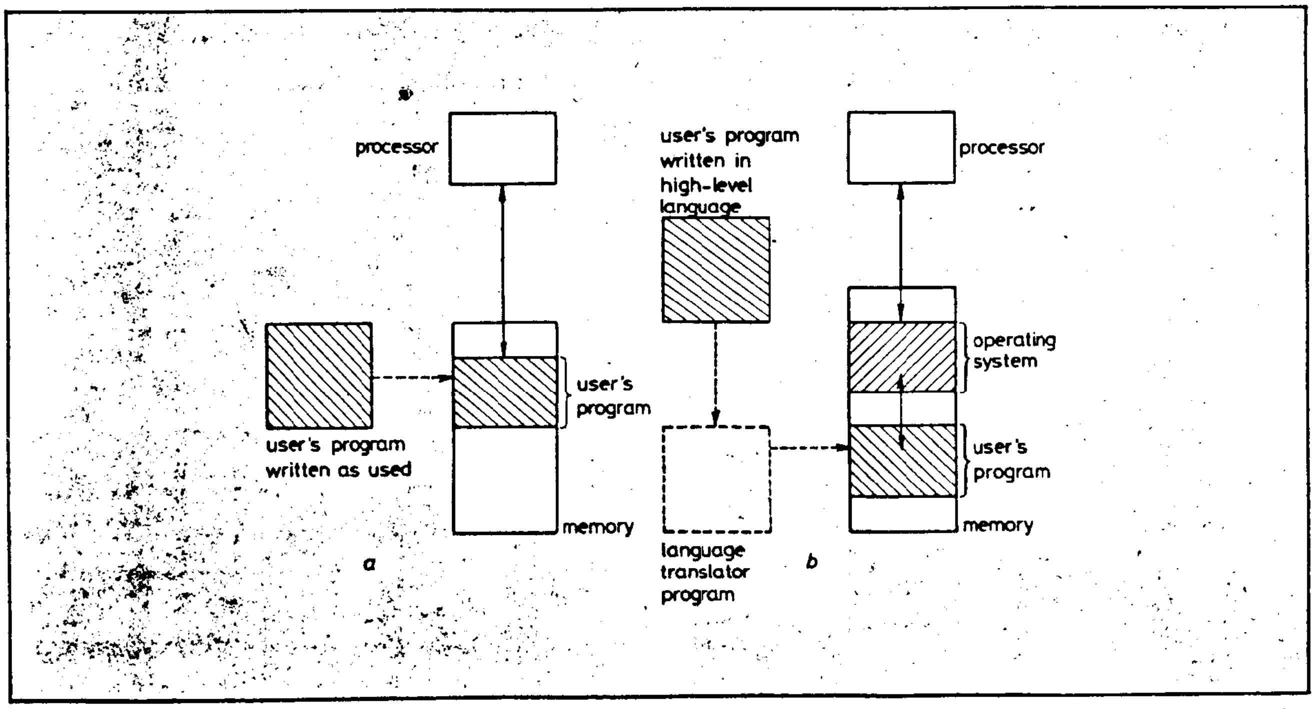
This shift in microprocessor design towards the implementation of more advanced architectural concepts is undoubtedly one of the most important developments in the short but already eventful history of the microprocessor. While the numerical dominance of the 4-bit and 8-bit logic-replacement devices will no doubt continue, and the second-generation 8-bit CPUs will certainly be around for many years to come, the new advanced-architecture 'supermicros' can be expected to exert a gradually increasing influence over the whole spectrum of microprocessor applications. This increasing influence will be the direct result of their advanced architectures, architectures based on concepts that, although often represented as being significant only in the high-end applications opened up by the greatly increased processing power of the 'supermicros', are likely to become recognised in the near future as just as significant for the less-demanding applications now in the province of the older second-generation devices.

This relevance of the third generation architectures to other than high-end applications has already been demonstrated to some extent by the introduction of the 'midi' microprocessors, devices such as the Intel 8088, Motorola 6809 and National Semiconductor 16008, which combine the implementation of advanced architectural concepts with the use of 8-bit interfaces that allow them to be used as replacements for the second-generation 8-bit devices with only a minimum of redesign. The scope for using these more advanced architectural concepts in the design of logic-replacement devices may be somewhat less, but important architectural developments can ultimately be expected in this area too.

Computer architecture

Before considering in detail the architectural advances embodied in the third-generation microprocessors, it may be best to specify more closely what characteristics of a computer-like device are included in the term 'architecture'. Essentially, by 'architecture' is meant the overall functional organisation of the device as seen from the applications viewpoint, i.e. the view of the device commonly taken by the user rather than the manufacturer.

Among the characteristics of a microprocessor that contribute to its overall architectural specification are: its instruction set; the data types on which its instructions operate; the



3 Modern approach to computer use. In the traditional approach (a), now rarely used in mainframe and minicomputer applications but still the norm for microprocessor use, the user programs directly in low-level hardware oriented code that runs unaided in the system. In the modern approach (b), the user programs in a high-level language and the translated low-level program drives the hardware via the operating system

number, bit-length and function of its registers; the addressing modes by which it can refer to specific locations in its memory; and the logical organisation of the input/output (I/O) units that it uses to exchange data with external equipment.

Concepts and features

Clearly, the concept of architecture is a highly multivariate one, and there is no simple measure of architectural sophistication, although several formal systems have been proposed. Clearly also, the desirability of certain architectural features will vary greatly with application, so that, for example, what is a good architectural arrangement for a logic-replacement device designed to control individual I/O lines will not be very suitable for a general-purpose CPU designed to interface with a system bus: for this reason there is little point in directly comparing the architectures of microcontrollers and general-purpose devices. In spite of these complications, however, it is possible to summarise concisely what a good architecture should do: it should allow all of the computing resources of the device to be utilised as effectively as possible.

Perhaps the best way of illustrating this concept of architectural 'goodness' is to consider the problems caused by architectural features that inhibit the effective use of a device's computing resources. Typically, this might be due to the lack of an instruction that performs a type of operation required by the application, or the lack of a data type required by the application, or, more subtly, the inability to use a particular instruction on an item of data solely because it happens to be located in one particular register. It is in the nature of computing devices that all these problems can be overcome — after all, any processor can eventually be programmed to do anything, but only at the cost of additional complexity in the software.

This additional complexity has two main consequences, one of which is that the additional operations required for the processor to 'work around' the architectural block can severely slow down the useful work of the application: improving the architecture of a processor can greatly increase its effective throughput even though the actual circuitry it employs continues at the same speed. Also, the additional complexity required in the software makes life much more difficult for the programmer, particularly if the architectural restrictions are themselves arbitrary and not easy to remember, and this leads to costlier and more error-prone software.

In case it should be thought that this effect is likely to be relatively minor, it should be remembered that programmers who have worked on both conventional minicomputers and second-generation microprocessors have often found that coping with the restricted architectures of the latter can almost double the time needed to complete a program.

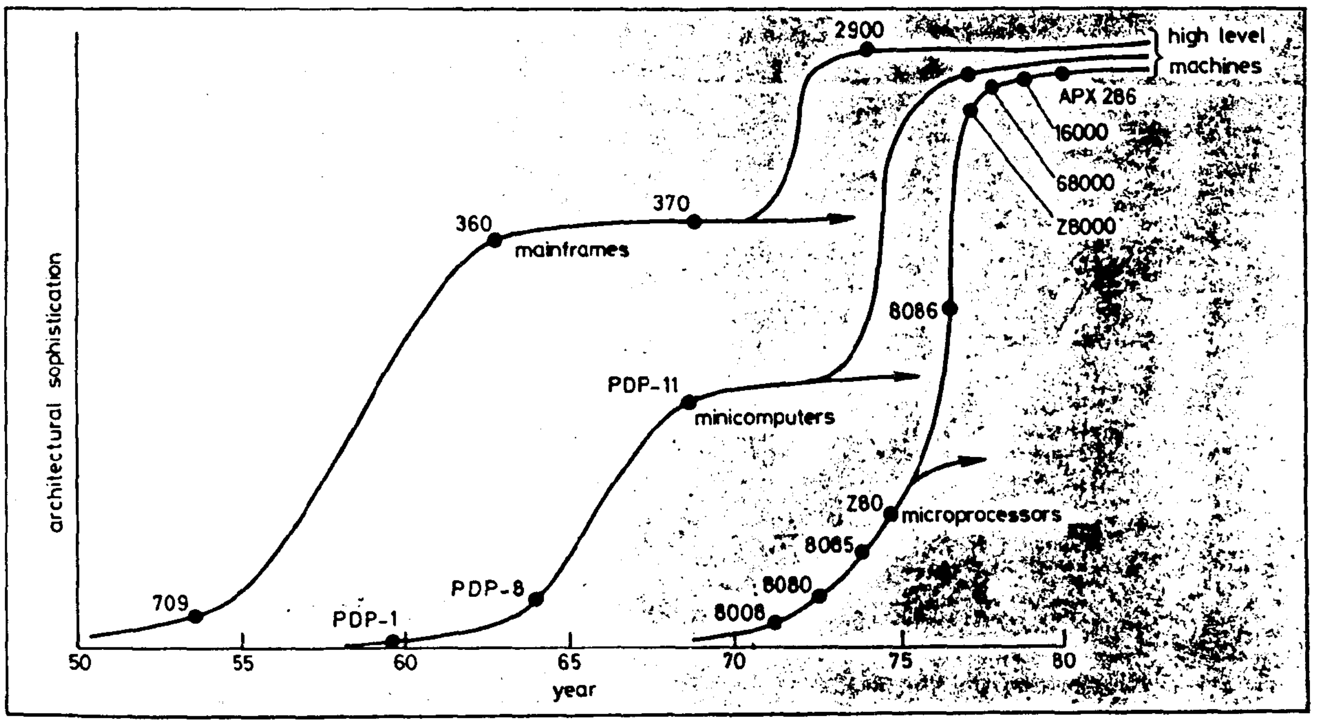
By the time the design of the new 'supermicros' came to be finalised, the fact that the second-generation microprocessors had such architectural deficiencies was well known to their manufacturers. The manufacturers were also well aware that their business was founded on their ability to provide their customers with readily usable computing power: there was clearly no advantage in selling devices that were unnecessarily difficult to use, and every possible advantage in selling devices that were easy to use and would therefore tend to be in greater demand.

Consequently, the decision was taken to use the greater capability of the HMOS technologies to remove as many as possible of the second-generation devices' architectural limitations. For example, the limitations on data types would be removed: instead of the typical second-generation limitation to just 8-bit data words (with 16-bit address pointers), the new-generation devices would directly support all the data types commonly used in microprocessor applications: bits, BCD digits, 8-bit data (bytes), 16-bit data, 32-bit data and variable-length character strings of practically unlimited (64k bytes) length.

Limitations

Similarly, the limitations on addressing modes would be removed: instead of the typical second-generation limitation to modes such as pointer, register, immediate and indexed, the third-generation devices would also allow many others, such as indirect, indirect indexed, base indexed, autoincrement etc. Also, more and longer registers would be provided, and care would be taken to make the instruction set as regular (or *orthogonal*) as possible, so that *any* instruction should, in principle, be able to operate on *any* data type contained in *any* register or in *any* memory location specified by *any* addressing mode.

In choosing to generalise the architectures of their future third-generation devices in this way, the microprocessor manufacturers were reacting to limitations in the architectures of their second-generation devices that had been identified as a result of practical experience in their application. However, they also



4 Enter the high-level architectures. The need to support high-level use of all types of computing device has led to the emergence of a new generation of high-level machines, developments of mainframes, minicomputers and microprocessors all based on very similar architectural concepts

had theoretical guidance in their choice as a result of a clear two-fold precedent for what they were doing, which had in fact been done before by the mainframe manufacturers at the beginning of the 1960s and by the minicomputer manufacturers at the end of the same decade.

The parallels between the evolution of these two older types of computer and that of the microprocessor are very noticeable (Fig.2) but this is not perhaps surprising: each of the three types of computing device has been brought into existence by the emergence of a particular form of electronics technology, and its evolution during at least the early years of its existence has been determined largely by the gradual maturing of that technology. Thus, mainframes were originally developed with discrete-component technology, and the early limitations on the components, first valves and then primitive transistors, meant that very unsophisticated architectures were originally adopted and that the rate of architectural advance was relatively slow. Only by the later 1950s had transistor technology matured to the point when hardware limitations were no longer a real restraint on architectural advance, and this then allowed the introduction of the first advanced-architecture main-frame, the IBM 360. Similarly, minicomputers based on SSI technology, started off with very basic architectures that continued until the maturing associated with the emergence of MSI devices, which then allowed the introduction of the first advanced-architecture mini, the DEC PDP-11.

Evolutionary pressures

In the case of the microprocessor, exactly the same evolutionary pressures were at work. First implemented in the relatively immature PMOS technology of the early 1970s, microprocessors were restricted by hardware constraints to very basic architectures, which in their broad characteristics were very similar to the mainframe architectures of the early 1950s and the minicomputer architecture of the early 1960s. Once the much more capable NMOS technology emerged, however, many of the hardware constraints were lifted, and microprocessors, like the mainframes and minicomputers before them, went through a stage of rapid architectural development, as typified by the transitions from the 8080 to the 8085 and then to the Z80.

Finally, with the then imminent commercial availability of the HMOS technologies, hardware limitations were no longer a real restraint to architectural advance, and the 'supermicros' were planned. It thus seemed likely that the evolution of the microprocessor was destined to go much the same way as the evolution of the earlier types of computer device, with the emergence of a definitive microprocessor architecture characterised by the provision of a substantial number of 16-bit or 32-bit registers, a useful range of data types and addressing modes, and a very regular instruction set.

Such a definitive architecture, or so it seemed at the time, would then 'fossilise' as a result of a number of stabilising factors, notably the large base of software that would quickly be built up for use with it, so forcing a virtual halt to major architectural development. After all, it was pointed out, very few new mainframe architectures have been developed since the introduction of the IBM360 and its 370 follow-up, and very few new minicomputer architectures had been developed since the emergence of the PDP-11.

Other factors were at work, however to prevent this coming about. One of these was the effect that LSI technology was having on all forms of computing: by dramatically reducing the cost of all hardware resources, particularly memory, the new technology was highlighting the restrictions of existing architectures as evidenced in unnecessarily high programming and usage costs. In addition, the progressive

Origins of the microprocessor

Intel's launch of the MCS-4 system just a decade ago certainly deserves recognition as the first commercial introduction of a microprocessor range, but in fact the 4004 was just one of three pioneering microprocessor devices that were developed almost simultaneously and launched within just a year or so of each other. The remarkable thing is that each of these three devices was the first example of one of the three very different types of microprocessor that have developed down to this day: the Intel 4004 was the forerunner of the microcontrollers and microcomputers designed for logic-replacement applications; the Intel 8008 was the first general-purpose single-chip CPU; and the National Semiconductor GPCP was the first of the bit-slice microprocessors. Behind the development of each of these devices lies an interesting story.

Behind the development of the 4004 lies the steady increase in the scale of microelectronics integration that continued throughout the 1960s. By the end of that decade, it was clear that the new LSI technologies offered remarkable benefits in a wide range of application areas, but only if the volume of devices demanded by the application was large enough to justify the high costs of 'custom' LSI design. For lower-volume applications, the problem was the now classic one of the lack of any really practical way of 'customising' mass produced, and therefore inexpensive, LSI devices to meet the specific needs of the application. Until such a way was discovered, the majority of users were effectively disbarred from exploiting LSI technology.

Ironically, the impetus to the solution of this problem by means of the programmable-logic LSI device or microprocessor resulted from the actions of a company that believed its volume of production could fully justify the use of custom LSI devices. The company was Busicom, a Japanese manufacturer that had designed a set of 11 low-density LSI devices to form the basis of a new product, a compact new desktop calculator. In spite of the fact that these devices were destined for use in a calculator, they were in no sense programmable, but just a collection of hardwired logic capable of handling the very simple 'four-function' calculations then required. Having designed these custom-LSI devices, however, Busicom then found that there was no Japanese semiconductor company capable of making them, a difficulty that it solved by approaching Intel in the USA.

Then a relatively young company, Intel was glad of the opportunity to manufacture Busicom's devices, but in the end decided that high-volume production of 11 different chips would tie up more of its manufacturing resources than it then wanted to divert from the production of its own designs. The solution seemed to be to redesign Busicom's logic to make use of Intel's relatively high-density PMOS technology and so make them fit on fewer devices. A young engineer called M.E. Hoff was then assigned to the project, and began working on the more general problem of how to design a set of readily customisable logic chips. Being from a computer-engineering background, Dr. Hoff eventually realised the similarity between what he was trying to do and what was done in computer design when a standard hardwired CPU was implemented instead by *microprogramming*, i.e. by using a basic but relatively fast 'computer within a computer' to perform the same functions as the hardwired CPU but with a greatly reduced amount of logic.

Using this analogy with microprogramming, Dr. Hoff then realised that the same approach could be used in the design of the Busicom calculator, with the original hardwired calculator logic being replaced by an appropriately microprogrammed 'computer within a calculator' built from just a few LSI devices. Equally well, any other suitable array of hardwired logic could also be replaced by exactly the same LSI devices, only the microprograms requiring any change. Within a few months, Dr. Hoff's ideas had been taken up within Intel, and the detailed design of the world's first logic-replacement microprocessor, later to be designated the 4004, had begun.

The development of the other two pioneering development of improved software techniques had made a re-orientation of hardware design overdue: existing architectures, whether mainframe, minicomputer or microprocessor, were all, to a greater or lesser extent, hardware-oriented designs geared to simple low-level programming techniques.

What was instead required were architectural designs geared to more modern techniques of programming and using computers, particularly the use of high-level programming languages and sophisticated operating systems, both of which act as 'bridges' between, on the one hand, the high-level approach to utilising computing resources that the user naturally wishes to adopt, and, on the other hand, the low-level approach actually allowed by conventional architectures. The use of such software 'bridges' to overcome the deficiencies of low-level hardware architectures can certainly be made to work, but usually only at the cost of providing very elaborate language-translation and operating-system software, and of accepting considerable inefficiency and general clumsiness in the overall operation of the system.

This need to develop new 'high-level' architectures, coupled with the availability of the new LSI technology from which new 'high-level' machines could be built, led to a rethinking of architectural concepts followed by development programmes dating from the beginning of 1970. In the mainframe world, these led most notably to the advanced architecture of the more recent Burroughs machines and of the ICL2900, and, although the mainframe industry is still dominated by the IBM 360/370 architecture there are signs of architectural change even in the IBM world. In the minicomputer world, perhaps the most notable development has been that of the DEC VAX-11, a 'high-level' extension of the PDP-11 that, among many other enhancements, takes the traditionally 16-bit minicomputer architecture up to 32 bits.

Supermicros

Finally, in the microprocessor world there have been the 'supermicros', devices that are based to a very large extent on the same 'high-level' architecture concepts as the new mainframe and minicomputer models. Indeed, reading the published design criteria for the new devices is rather like reading slightly adjusted versions of the design criteria published for machines such as the ICL2900[3] and DEC VAX-11[2]. This similarity is most notable with the later 'supermicros' the Z8000, 68000 and 16000, which at the time of their respective introduction each displayed progressively more similarity to the VAX-11 design in particular.

As an example of the high-level facilities provided by these new advanced-architecture microprocessors, consider National Semiconductor's remarkable new high-end device, the 16032. This powerful CPU, supported as desired by coprocessors for floating-point arithmetic and memory management, is essentially a 32-bit machine with eight 32-bit general-purpose registers, eight 32-bit floating-point registers, and six 24-bit pointer registers. With its 24-bit addresses, the device can address directly 16Mbytes of memory (larger memories based on 29-bit addresses will be supported in later releases), and it also includes facilities that will allow full virtual-memory operation[4]. Memory can be addressed by eight different addressing modes, including all those usually found on advanced-architecture devices, and the data types include bits, bit-fields, BCD digits, 16-bit words, 64-bit double words, single-precision and double-precision floating-point numbers, and variable-length strings. The instruction set includes over 100 basic types, and is highly orthogonal. In all these respects, the device shows an architectural sophistication up to traditional mainframe levels.

More than this, however, the device

---

devices was carried out for different reasons: the aim in both cases being not to find a way of replacing generalised hardwired logic with 'customised' LSI, but to adapt the newly emerged LSI technology to the needs of minicomputer manufacturers. Ideally, these manufacturers would have liked to have been provided with a single-chip LSI device having the same characteristics, including processing power, as their existing MSI CPUs, but everyone believed, rightly as it turned out, that this would not be possible until well into the 1970s. The alternative approach, as adopted by engineers at National Semiconductor, was to try to develop a set of LSI components that, while not being full CPUs in themselves, could be used in relatively small numbers to construct a CPU according to each individual manufacturer's specific requirements. What the National Semiconductor engineers were looking for, in fact, was a very versatile set of LSI devices that could be used to build a readily 'customisable' minicomputer CPU.

Again, the solution to this problem was found in the existing concept of microprogramming. By designing a set of LSI devices that could form a microprogrammed 'computer within a computer', National Semiconductor realised that it could provide users with all the benefits of LSI in a form that only required suitable microprogramming to allow full customisation. The limitations of the then PMOS technology, coupled with the need to optimise the devices for speed in order to attain the required minicomputer-like performance, meant, however, that several devices would be necessary to provide even the nucleus or 'microinstruction execution unit' of a CPU, a further problem that received an elegant solution in the form of the 'bit slice' concept.

Following this concept, an assembly consisting of only four GPCP 4-bit 'slices', 15 - 20 TTL support devices and a ROM to hold the microprograms, could form a complete minicomputer CPU, a point that National Semiconductor clearly demonstrated by launching a pre-assembled model called the IMP-16, the first single-board minicomputer. Although the bit slices could be used to build lower-performance 4-bit and 8-bit processors, they remained identified with 16-bit minicomputer-like systems, and this led National Semiconductor to go on to develop single-chip versions of the IMP-16 such as the late first-generation PMOS PACE, and the second-generation NMOS INS8900.

Even while the GPCP bit slices were still being developed, however, another project aimed at bringing LSI technology to computer design was under way. The aim of this other project was crucially different, however, for it was meant to result not in an LSI-based CPU of minicomputer-like performance, but in a specially low-powered CPU for a specific range of low-end applications, actually for relatively simple local processing in 'intelligent' terminals. The idea of using a low-power 8-bit LSI CPU for such applications was first promoted in 1968 by a small company called Viatron, which unfortunately went bankrupt before its development project was complete. The idea was revived, however, by Datapoint, a computer company then as now specialising in 'intelligent terminals' and what has more recently become known as 'distributed processing'. Datapoint approached Intel with its ideas, and a project was started with the aim of integrating as much as possible of a low-power 8-bit CPU onto a single chip.

In fact, it was only because a specifically low-power CPU was being built that this project had any hope of success: if minicomputer-like performance had been required, the Intel engineers could have had to resort to the sort of multichip solution that had been chosen by their National Semiconductor colleagues. As it was, the limitations of the then available PMOS technology meant that not all of even a low-power CPU could be squeezed onto a single chip, and the resulting device had to be supported by usually some 20 - 40 TTL devices to give a full-function CPU.

Nonetheless, the new device, designated the 8008, was much nearer to a single-chip CPU than many had at the time thought possible, and its capabilities were enough to establish it and its second-generation successor, the 8080, as almost the definitive general-purpose microprocessor.

DENNIS MORALEE

also provides specific hardware support for high-level operation. The use of operating-system software, for example, is aided by the provision of two levels of operation, the normal level used by the users' programs, and a supervisory level used by the operating software. User programs executing at the normal level are debarred from using many of the device's instructions, so that all operations that will crucially affect the state of the system have to be carried out by passing control to the operating software, which is thus able to supervise at all times the overall operation of the system.

Similarly, the efficient execution of code generated by translator software from programs written in high-level languages is promoted by an instruction set geared to this task. In particular, the distinctive 'block' structuring of programs written in modern high-level languages such as Pascal and Ada is directly supported by sophisticated ENTER and EXIT instructions, each of which is equivalent to a series of several instructions in conventional architectures. Also, a series of special instructions and other facilities directly support the use of independently constructed program modules, with the links between modules being handled by the processor's hardware without the need for complex and performance-limiting linking software.

Facilities such as these mean the removal of many of the limitations that in the past have made programming microprocessor systems unnecessarily difficult. The high speed of the HMOS third-generation devices, their more efficient architectures, and the availability of coprocessors and other multiprocessing facilities, all add up to a practical removal of all computing-power limitations for the great majority of applications. Similarly, the large addressing range of the third-generation devices, their use of memory-management techniques, and the possibility of virtual-memory operation, all add up to a practical removal of all memory-size limitations.

Finally, the ability of the devices to efficiently execute machine-translated code, their direct support of the most commonly used data types and structures, and their suitability for use with sophisticated operating software designed to take responsibility for detailed control of system resources away from the user, all add up to a removal of the practical limitations on high-level programming and operation. Since it has proved to be the limitations on computing power, memory size and level of programming imposed by the second-generation devices that have created most of the difficulties in using them, the removal of these limitations in the case of the third-generation devices can be expected to make them very much easier to use.

In the wider context of the evolution of computer architectures, what the introduction of the advanced-architecture third-generation devices means is that the microprocessor manufacturers have effectively made two significant jumps in architectural sophistication at the same time. The first jump is that from the restricted and irregular architectures of the second-generation devices to the more capable, more commodious and more regular architectures typical of such successful if now ageing computer designs as the IBM 360 and PDP-11. The second jump, made simultaneously with the first, is from these enhanced but still hardware-oriented architectures to the very advanced 'high-level' architectures of the third-generation devices.

By making this twofold jump in architectural sophistication in the last few years, what the microprocessor manufacturers have managed to do is to close the gap between themselves and the mainframe and minicomputer manufacturers. Now, although there are still major generic differences between the mainframe, minicomputer and microprocessor realisations of the high-level architectural concepts, no on[e] can any longer say that the microprocessor devices are architectura[lly] much less sophisticated than their mainframe and minicomputer counterparts.

Full potential

This greatly increased sophistication of the new-generation microprocessors is bound to be reflected in a corresponding increase in the sophistication of the new products that will be based upon them. At the moment, products based on the pioneering if relatively low-end 8086 are becoming fairly common, while products based on the newer Z8000 and 68000 devices are just beginning to make an appearance. It seems safe to say, however, that none of the products yet introduced manages to exploit anywhere near the full potential of the third-generation devices, which will only be fully realised in product form much later into the 1980s.

One restricting factor here is the very magnitude of the change from the second-generation to the third-generation devices, which means that users have a great deal to assimilate before they can be expected to use the new devices in the high-level way intended by their designer. High-level facilities may indeed be easier to use, but, to the unprepared user at least, they are not necessarily easier to understand.

The development of the 'supermicros' has thus provided a firm basis on which the user community can be expected to build well into the 1980s, gradually incorporating more and more of the devices' increased sophistication into their products. What will be the full effect of this increased sophistication on the end-user markets is hard to imagine — after all, it would be difficult to say that the full impact of even the second-generation microprocessors has yet been felt by the end-users of electronics-based equipment. Nevertheless, it now seems at least possible that, even before the advanced features of the third-generation devices have been fully assimilated by those who will eventually exploit them, the first example of a yet more advanced fourth-generation of microprocessors has been introduced.

This yet more advanced device is, of course, the new iAPX 432, which may prove to be, after the 8008, 8080 and 8086, yet another new-generation 'first' for Intel. What the 432 represents is not just another triumph for Intel's microelectronics technology — the three chips making up the 432 range contain some 225 000 transistors compared to the 2250 in the 4004, an increase of 100 times in just 10 years — but also another example of its willingness to incorporate into its products the most advanced concepts available from the user community. Although many, but not all, of the 432's features have been discussed in computer-science circles for some years, sometimes built into research-oriented machines, and occasionally implemented in a partial way in some of the most advanced computing systems, usually mainframes, to reach the market, there is no doubt that, for a mainstream product soon to be available in high volume, it represents probably the most sophisticated general-purpose computing device yet built.

As an example of this sophistication, the 432 implements a version of the 'capabilities' approach to memory management, an approach formerly limited to a few research-oriented machines. This approach effectively limits access to each item of data in memory on a 'need to know' basis, the hardware itself checking each attempted access to ensure that it is legal, an arrangement that can do much to detect, contain and even correct execution errors. Memory accesses are also made on a 'descriptor' basis, in which programs access data not by an address but by what is effectively a description of what the data are, the actual accessing then being carried out by the hardware according to its own 'knowledge' of where the data are stored in gigantic 2^{40} byte virtual-memory space. The hardware also monitors the 'type' of each data item, thus for the first time supporting one of the more powerful features of the most modern high-level languages.

It is interesting that all these features — data typing, descriptor addressing and 'capabilities' phased memory management — were actually considered by the VAX-1 project team in 1976, but were in fact rejected. The reason given for rejecting the 'capabilities' approach is particularly significant: 'the complexity of the capabilities design was inappropriate for a minicomputer system'.[2] Few would have disagreed with this at the time, but even fewer would have dreamt that only 5 years later the approach would be implemented on a microprocessor.

Just as remarkable is the 432's approach to supporting high-level operation, via both high-level programming languages and sophisticated operating