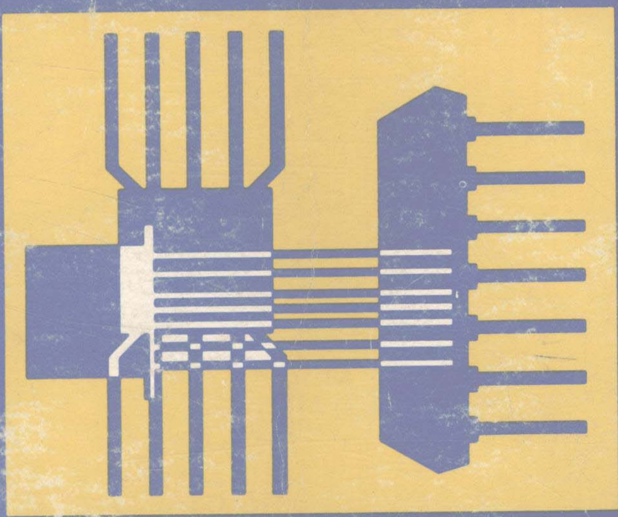
microprocessors

NEW DIRECTIONS



FOR DESIGNERS

Edited by Edward A. Torrero

Selected from ELECTRONIC DESIGN

HAYDEN

MICROPROCESSORS

New Directions for Designers

Selected from

ELECTRONIC DESIGN

Edited by
EDWARD A. TORRERO
Associate Editor, *Electronic Design*

HAYDEN BOOK COMPANY, INC.
Rochelle Park, New Jersey

Library of Congress Catalog Card Number: 75-21855

Printed in the United States of America

9 PRINTING

77 78 79 80 81 82 83 YEAR

MICROPROCESSORS

New Directions for Designers

Selected from

ELECTRONIC DESIGN

Preface

Digital system designers now have a remarkable new tool. Microprocessors can lower the cost and increase the flexibility of electronic equipment, thereby blazing a trail for new and exciting uses.

Applications once confined largely to point-of-sale-terminals now run the gamut from traffic-control systems to small accounting equipment, and from computer terminals to industrial-process controllers. In some cases, microprocessor-based systems have even replaced dedicated minicomputers.

Together with memory and peripheral circuitry, processor chips form complete microcomputers. In complexity, these micros fall midway between small, hand-held calculators and conventional minicomputers. And microcomputers bring with them some of the features of both. Like calculators, they're compact and inexpensive. But like minicomputers they can be programmed for a wide range of tasks and work with such computer peripheral devices as magnetic memories and high-speed printers.

Through imaginative design, engineers can use software to adapt basic microprocessor hardware for a host of applications. However, the software phase of design can entail an extensive learning process for those engineers who are more at home with gates and registers.

This book deals with both the hardware and software aspects of microcomputer design. It presents application articles, tutorial discussions, and survey reports published in *ELECTRONIC DESIGN* from 1973 to 1975. The emphasis is a practical one. Subjects covered in detail include how to select circuits, how to interpret their capabilities, how to extend their useful range, and how to apply them.

July 1975 EDWARD A. TORRERO

Contents

SECTION I:

Industry Enters the Microprocessor Era

The first article discusses the wide range of products available and explores the pitfalls of microprocessor selection. Other articles deal with microprocessor penetration into instrumentation and industrial electronics, and its threat to minicomputers.

Focus on Microprocessors

EDWARD A. TORRERO
Associate Editor, Electronic Design

Digital designers faced with the many performance claims for new LSI microprocessors might conclude that one of these little, low-cost marvels could solve all their system problems. After all, if a small IC offers the processing of a computer, replaces scores of standard logic circuits, and has a seemingly endless list of applications, who needs to design with anything else?

Closer examination of the burgeoning application literature for a "computer on a chip" reveals a different picture. A large-scale integration processor, for example, does perform many of the functions of the central processing unit in conventional computers. But to use the circuit, many more ICs may be needed to interface with peripheral devices, data-communication lines and even its own memory.

And that low price tag on the LSI processor—typically, well under $100 for 4-bit word-length units and under $400 for 8-bit units—is low compared with the thousands of dollars needed for a general-purpose minicomputer. However, to build a full-fledged microcomputer, you need memory, and the cost of that can easily exceed the price of the processor.

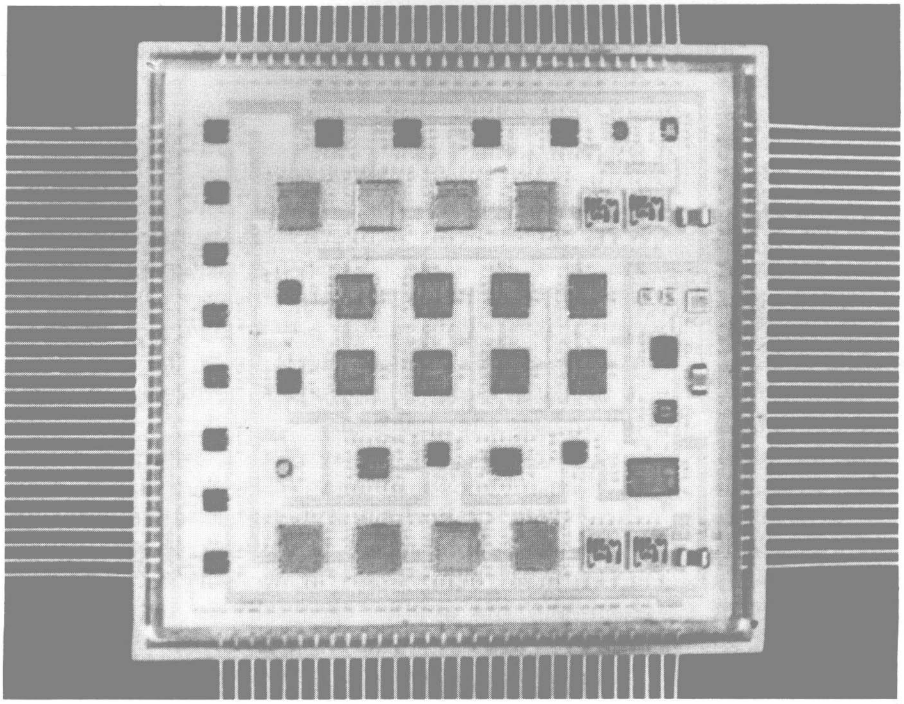
Moreover, if you add up the cost of all the necessary hardware components, you may find the total exceeding the less-than-$1000 level of "stripped down" minicomputers available on PC boards.

Combine an LSI microprocessor with associated circuitry and memory, and you obtain a microcomputer. In the foreground are LSI circuits and prototype systems offered by Rockwell International. The manufacturer's line includes 4 and 8-bit microprocessor chip sets.

Programming—unfamiliar demands

A decision to use an LSI processor opens up a whole new design ball game. Not only must a digital designer contend with the relatively familiar requirements of any logic system. The designer must also grapple with the relatively unfamiliar demands of programming. And software development, usually involving assembly language (only one step up from a computer's inherent machine language of ONEs and ZEROs), represents by far the major design effort and cost.

In addition microprocessors don't lend themselves to the traditional "learning curve" for new



Dice mounted on a substrate form a microprocessor system. This unique packaging approach is used by Teledyne Semiconductor. Other vendors typically mount DIPs on PC boards.

components. Previously the experience gained in the use of one component could be transferred to a similar component from another manufacturer. But different microprocessors generally don't have alternate sources. And the chips come with different software capabilities, hardware requirements and design aids. Hence a completed design, using one LSI processor, might have to be scrapped totally if you decide to turn to another vendor's unit.

Still, there are major benefits. Microprocessors are unique among ICs in that they can be programmed like computers. As a result, they permit a tradeoff of software for hardware to achieve a dazzling increase in system capability and versatility.

They can be used economically to replace or upgrade random-logic designs, involving scores of standard ICs, when many functions must be performed. And they use less circuitry than hardwired logic in applications emphasizing random collection and routing of data.

Of course, for some applications microprocessors are not the sole LSI alternative. Complex logic decisions might be handled just as well with circuitry using programmable logic arrays. And arithmetic computations are performed by arithmetic logic units or by calculator chips—from which a number of microprocessors have evolved. Custom LSI chips provide yet another alternative.

Nevertheless microprocessors are filling the gap between special-purpose LSI circuits and conventional minicomputers. They are currently finding their greatest use as decentralized, cheaper minis in remote, programmable controllers.

There's a wide product range

The many advantages of microprocessors are creating a demand that manufacturers are meeting in a variety of ways. Designers can choose from among single or multichip processors, chip sets or PC-board assemblies, and from among a wide range of technologies.

The most common technology is silicon-gate, p-channel MOS (PMOS). But manufacturers are also using n-channel MOS (NMOS) and silicon-on-sapphire MOS (SOS/MOS) to achieve speeds that are higher than those possible with PMOS. Bipolar processes are employed for the highest speeds—about 200-ns cycle vs 2 μs for NMOS types. Complementary MOS (CMOS) is used for the lowest power dissipations—microwatt-range chip dissipation vs milliwatt range for other types.

Standard LSI-processor products in various configurations handle data in 4, 8 and 16-bit word lengths, and modular multichip microprocessors can be used to achieve even longer word-length processing. The available configurations, with their major features, consist of the following:

- Microprocessor chip sets, including special interface ICs and sometimes special memories, to simplify designs of minimum-hardware systems for specific applications.
- Microprocessor-based logic boards to eliminate the need to test, assemble and interconnect

processor chips, peripheral circuits and memory for a variety of applications.

▪ General-purpose—nondedicated—microcomputers, on cards or in boxes, to permit system design, development and testing. These are offered by component manufacturers, as well as, a growing number of other vendors.

▪ Microprocessor-based minicomputers offering, as a result of their traditional mini features, maximum flexibility and capability when compared with MOS microcomputers. Offered by minicomputer manufacturers, these units generally use custom MOS processors.

The cost of each configuration increases with a unit's complexity. At the minicomputer level, hardware cost might be the highest. But available software support is the most extensive. Designers tend to feel that the number of units determines the major tradeoff in a choice between a micro and minicomputer. A small number of units—as for an end-user application—can best be served by a mini, which can minimize software development costs. But for large quantities—as for an OEM application—a microcomputer can minimize hardware costs.

Chip sets improve early versions

Due to increasing availability of special interface circuits and improvements in processor-chip architecture, fewer additional circuits are needed for the newest microprocessors than for earlier versions. First-generation 8-bit processors, for example, typically needed about 20 additional standard-TTL circuits to make them work.

The extra ICs include the following: registers to address memory, either ROM or RAM; decoders to interface with memories; other ICs to handle processor information and to synchronize the operation of the processor and circuits; clock circuits, and a variable amount of interface ICs, depending on the application. For example, in a multichannel data-communications application each channel requires an asynchronous receiver/transmitter and associated interface ICs.

But even with newer processors, applications still can require additional circuitry to obtain one or more of the following: clock generation and timing, memory and I/O control, data and address buffering, multiplexed inputs, interrupt control, refresh for dynamic memories and additional supply voltages.

And some microprocessors are offered only as part of complete chip sets or with the purchase of the associated memory from the same IC manufacturer. This may not be a problem if you plan to buy all your components from the same source, but it does stop you from shopping around for the lowest price and precludes the use of core memory—which most processors can accept just as well as semiconductor types. Also, if you purchase a chip set, you must design around the circuits offered.

Specs don't tell all

Unlike the less-complicated ICs, microprocessors cannot be completely characterized on a simple data sheet. Moreover different vendors use different parameters to measure a processor's capabilities. This makes any comparison of processors—not to mention selection of the best one—a difficult task. And there's no trend in sight toward standardization.

A microprocessor's computing speed is a case in point. Frequently manufacturers use a basic cycle time, or period—sometimes called a microcycle—to denote speed. But many microcomputer operations require several such cycles to be performed. This applies especially to the execution of the more powerful instructions. Hence a critical instruction may require more time than that indicated by the basic cycle.

In addition a microprocessor's maximum clock rate can be misleading, if taken for a measure of speed. It's possible for one microprocessor to perform basic operations—like register-to-register add—faster than a unit using a higher clock rate. Differences in microprocessor architecture and chip design tend to minimize the importance of the clock-rate spec.

Other specs given to indicate speed include minimum instruction time, interrupt response time, and time to add two numbers—which may already reside in the processor. Like cycle time and clock rate, these numbers don't measure such critical times as the over-all time needed to perform important routines. Excluded are additional delays, such as those needed to obtain data from memory. The solution is to use benchmark programs tailored to your application as a basis for selection of one microprocessor over another.

Use of benchmark programs can determine the power of the instruction set, thus circumventing one of the most abused specs—the number of instructions. Microprocessor comparisons based on this number abound, although such comparisons have serious flaws.

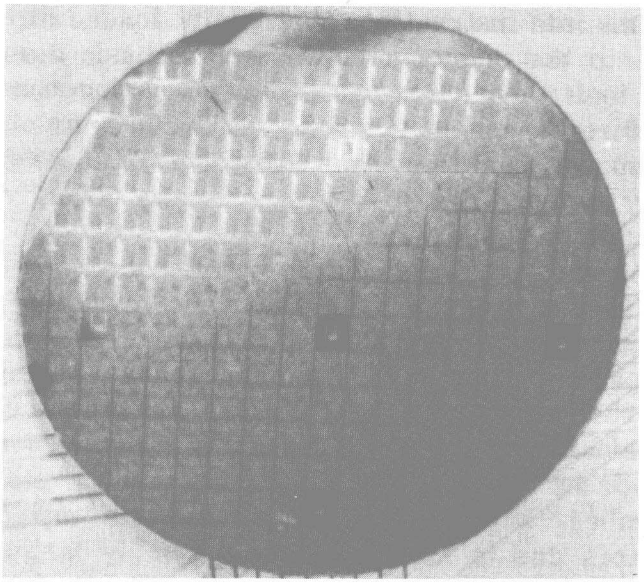
For example, a simple number doesn't reveal what instructions are available for data movement and manipulation, for decision and control and for input/output operations. Some microprocessors have far more I/O instructions than others; they are tailored for a specific class of applications. And missing instructions can always be performed by routines, although with a sacrifice in speed.

Also, the number of instructions claimed for the same microprocessor can increase from one page of a reference manual to another. One reason is that instructions that move data have been multiplied by the number of addressing modes.

Common addressing modes include direct, immediate and indirect. In the immediate mode, the instruction includes data, while in the indirect mode, an address preloaded into a register increases the address bits in an instruction. Variations and extensions of these modes are also available, so a basic instruction can be multiplied several times.

Other factors that inflate the number of instructions may be the number of registers—in, say, a load-to-register operation—or the number of conditions—for example, those on which a branch may occur.

Of course, improved instruction sets are obtained with longer word-length microprocessors and advanced versions of smaller units. For example, 16-bit microprocessors have instructions for multiply and divide—functions that require software routines in 8-bit models.



They're not "computers on a chip"—yet. But LSI microprocessors, symbolized by this wafer from Intel, perform many of the functions of central-processing units in conventional computers.

And advanced microprocessors feature more powerful instructions as well as the original set of a predecessor. However, this "software compatibility" doesn't allow routine upgrading of systems by chip replacement. In general, expect to redesign to employ the new hardware/software tradeoffs efficiently.

For virtually all models, data sheets claim TTL compatibility. But don't expect many MOS processors to drive TTL loads; most don't. The term primarily refers to the fact that both microprocessor and TTL circuit can use a common 5-V supply. Newer models list a maximum TTL-drive capability of only one standard load, and many models require external components to achieve the interface levels needed for logic compatibility.

Watch architectural claims, too

Many computer-like features of microprocessors are frequently cited, including the number and function of registers, the type and depth of stack, interrupt capability and direct-memory access (DMA). However, there isn't as much architectural diversity as there is with minicomputers. IC manufacturers are constrained by technology limitations, so that comparable microprocessors tend to perform similarly. For example, preliminary benchmark programs run by several manufacturers for their 8-bit, single-chip NMOS processors often show comparable execution times.

Data sheets frequently boast several working registers. But only a single register, the accumulator, is essential. An accumulator, however, must have access to memory, and available instructions should permit immediate addressing and data manipulation between the accumulator and memory. If indirect addressing is available, even the function of special index registers can be accomplished with memory.

The major significance of additional registers lies in access time and the bit efficiency of instruction words. It takes far fewer bits to specify one of several previously defined working registers than a memory location. And a faster execution time can be obtained with registers that are separate from memory. They can be accessed without excessive memory-cycle delays. Otherwise it doesn't matter whether these registers are in an external memory or in the processor, so long as they can be referenced efficiently.

Some data sheets might seem to imply that the quantity of registers is more significant than their quality. It isn't. Not all registers can be incremented and tested for zero, even though they are described as "general-purpose." Those that can may be used for counting and program-loop control. In general, not all registers can be used for indexed addressing. Nor can they be loaded directly from memory—rather than only from the accumulator—or used as a source or destination for arithmetic logic operations.

Microprocessors employ stack-oriented registers that can be accessed only in a last-in-first-out basis—the so-called LIFO, or push-down, stack. These are used for subroutine nesting, interrupts and for temporary storage of data. They can be either on the chip (a hardware stack) or external to the processor in memory (a software, or pointer, stack). The hardware stack permits higher-speed operation, but it has limited size. The size of the software stack may be as large as available memory space permits, but the stack must be maintained by the program.

An interrupt capability is an absolute must for applications that involve asynchronous or unpredictable events. All microprocessors claim some type of interrupt handling ability, but the extent can vary from one unit to another. With older processors, you have to design the means to save the contents of the processor just prior to the interrupt, and then restore the information after the interrupt is serviced.

Those means may involve reservation of registers on the chip, use of external registers or use of another microprocessor. Any of these methods can store the essential contents of the processing unit. Software control, involving special routines, must also be provided to complete the design. The complexity of these techniques tends to discourage designers from attempting to handle even a single interrupt.

Newer microprocessors can accommodate single-line, multilevel and vectored interrupts, and they save essential registers automatically. A complete saving must be programmed. In one single-line interrupt system, device-interrupt requests are ORed together to form one request line. The program identifies the device and resolves priority. A multilevel scheme employs several single-level sense lines to handle additional interrupts. For very fast response, the vectored interrupt directly branches to a memory location that corresponds to a specific interrupt.

Another feature that depends on the unit is DMA capability. For some units, a "direct" access of memory must be performed indirectly through the microprocessor's usual word-by-word transfer procedure. This may not be a problem if you don't mind the processor's idle time, but it does limit data-transfer efficiency.

And don't expect I/O data throughput rates always to include the time needed to sense for a device or to respond to an interrupt. When these times are included, the actual throughput can be significantly less than expected.

Fixed instruction vs microprogram

Most microprocessors come with fixed instruction sets, around which software must be developed for an application. For some units, however, the option exists for microprogramming, the ability to alter or totally change the original instruction set. In essence, you program the microprocessor's internal microinstructions to obtain a macroinstruction set that is tailored to the application.

The advantages of microprogramming include increased speed, since microinstructions are executed considerably faster than macroinstructions are. Also, the technique allows a more detailed level of control that can be used to reduce hardware; the program controls more functions.

Because of the hardware savings, vendors expect microprogramming to find its greatest use in large-volume applications—in excess of tens of thousands of units. Alternatively, microprogramming represents the logical choice for an emulation of another computing system or for the speedy execution of critical, short routines.

The exceptional skills required for microprogramming constitute its major disadvantage. A microprogrammer must deal with the specific timing relationships of the internal architecture. And since each application requires a separate microprogram, each has its own instruction set that can't be transferred easily to another application. Nor can software design aids, geared toward the fixed instruction set, be applied to the changed set.

Design aids speed development

Much of the start-up, or development, effort in the design of a microcomputer system is linked to the coding phase. Coding converts system programs into instructions that can be loaded directly into the memory. However, the basic design-aid tools themselves are programs that generally require the use of time-sharing services or other computer facilities. And like the LSI processors they support, design-aid features can differ from vendor to vendor.

Assemblers are a case in point. All assemblers convert a program into the basic machine language in a process that usually involves several steps. Essentially the assembler reads a so-called source tape—with statements written in the mnemonic, or symbolic, assembly language—and produces a so-called object tape, with binary numbers suitable for the processor's memory. Errors due to misuse of the assembly language can be detected and pointed out by the assembler.

But some manufacturers offer single-pass assemblers, thereby reducing the steps needed to obtain the binary instructions for memory. Or they may provide the option of loading the assembler into ROMs and pROMs so that the microcomputer itself, rather than a host computer, executes the program. These are called hardware assemblers.

Another type, called a macro-assembler, simplifies coding when similar sections of code are used repeatedly, but variations preclude the use of conventional subroutine techniques. With a macro-assembler, a single instruction yields the necessary expansion.

Editors, available on time-sharing services, allow designers to prepare the original assembly-language programs and to change or correct them with simple commands. They can add documentation and store, combine and retrieve programs. And they can readily output programs onto paper

MICROPROCESSOR SCORECARD®

μP Scorecard®	Classification	Technology	Parts Family: Clock Driver	Parts Family: I/O Interface	Parts Family: UART/USRT	Parts Family: RAM	Parts Family: ROM/PROM	Parts Family: Interface	Features: Interrupts	Features: Integrated CPU	Features: Microprogrammed	Features: Accessible Stack	Features: DMA Ability	Features: BCD Arithmetic	Word Size (Data/Program)	Address Capacity (Program Words)	Clock (KHz/Phases)	Register Add Time (μsec per Data Word)	Number of CPU Registers: ALU	Number of CPU Registers: XR	Number of CPU Registers: GP	Return-Stack Size (NR x Bits)	Voltages Required	Power Dissipation (Watts)	Operating Temperature Range (°C)	Package Sizes (DIP Pins)	Price Range (approx.; CPU only; Quantity 100)	Status	Remarks
BURROUGHS **MINI-D**	One-chip CPU with ROM	PMOS								✓					8/12	256	1000/1	9	3	—	1		−12, +5			16	$ 60	Custom	
FAIRCHILD **PPS-25**	Calculator-oriented	PMOS		✓		✓	✓							✓	4x25/12		400/2	62.5	1	—	—	4x12	−10, +5	.6	0, +70	16, 18 24, 40	$ 60	Delivered	
INTEL **MCS-4/4004**	Calculator-oriented	PMOS		✓		✓	✓	✓		✓				✓	4/8	4K	740/2	10.8	1	—	16	4x12	−10, +5	1.0	0, +70	16	$ 30	Stocked	
INTEL **4014**	Calculator-oriented	PMOS		✓		✓	✓	✓	✓	✓				✓	4/8	8K	740/2	10.8	1	—	24	8x12	−10, +5	1.0	0, +70	16, 24		Rumored	
INTEL **MCS-8/8008**	One-chip CPU	PMOS							✓						8/8	16K	500/2	20	1	—	6	8x14	−9, +5	1.0	0, +70	18	$100	Stocked	
INTEL **8008-1**	One-chip CPU	PMOS							✓						8/8	16K	800/2	12.5	1	—	6	8x14	−9, +5	1.0	0, +70	18	$130	Stocked	
INTEL **8080**	One-chip CPU	NMOS	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	8/8	64K	2083/2	2	1	—	6	(RAM)	−5, +5, +12	1.0	0, +70	40	$200	Delivered	
INTERSIL **ISD-8**	One-chip CPU	CMOS							✓	✓					12/12	4K	2000/1	6	1	—	—	Modifies Program	5	.002	−55, +125	40		Announced	DEC PDP-8 Code
MOTOROLA **6800**	One-chip CPU	NMOS		✓	✓	✓	✓		✓	✓				✓	8/8	64K	1000/2	2	2	1	—	(RAM)	5	.25	0, +70	24, 40	$150	Samples	
NATIONAL **GPC/P**	4-bit Slice	PMOS									✓	✓	✓		4N/23	100	715/4	1.4	8	—	—	16x4N	−12, +5	.7	0, +70	22, 24	$150	Delivered	1 ≤ N 5 6
NATIONAL **IMP-4**	3-chip CPU	PMOS						✓	✓		✓	✓		✓	4/4	64K	500/4	12	4	—	—	7x12	−12, +5	1.0	0, +70	22, 24	$150	Samples	16x4 Data Stack
NATIONAL **IMP-8**	3-chip CPU	PMOS							✓		✓	✓	✓		8/8	64K	715/4	4.6	3	1	—	16x8	−12, +5	1.0	0, +70	22, 24	$230	Delivered	
NATIONAL **IMP-16**	5-chip CPU	PMOS							✓		✓	✓	✓		16/16	64K	715/4	4.6	2	2	—	16x16	−12, +5	1.4	0, +70	22, 24	$310	Delivered	
RAYTHEON **RP-1600**	4-bit Slice	Bipolar									✓				4N/48	64K	5000/1	1	j	k	—							Announced	$(j + k) \leq 8$
RCA **COSMAC**	2-chip CPU	CMOS							✓				✓	✓	8/8	64K	667/1	6	p	q	—	r x16	12	.01	−55, +125	40		Announced	$(p + r + 2q) \leq 15$
ROCKWELL **PPS-4**	Calculator-oriented	PMOS	✓	✓		✓	✓			✓		✓	✓	✓	4/8	4K	200/2	5	1	—	1	3x12	17	.225	0, +70	42	$ 40	Delivered	
ROCKWELL **PPS-8**	One-chip CPU	PMOS	✓	✓		✓	✓		✓	✓		✓	✓	✓	8/8	16K	256/2	12	1	1	2	2x14	17	.3	0, +70	42		Announced	
SIGNETICS **2650**	One-chip CPU	NMOS							✓	✓				✓	8/8	32K	1200/1	4.8	s	t	—	8x15	5	.5	0, +70	40	$100	Announced	$(s + t) \leq 6$
TOSHIBA **TLCS-12**	One-chip CPU	NMOS		✓		✓	✓		✓	✓	✓				12/12	4K	1000/3	13	1	—	—	(RAM)	−5, +5	.8	−20, +80	16, 24 26, 42		Announced	Multiply Instruction

July 1, 1974

A tabulation of representative LSI microprocessors reflects the product range: from calculator-oriented, 4-bit units using PMOS technology to newer 16-bit multichip bipolar processors. Note that the table, reprinted from "New Logic Notebook" by Microcomputer Techniques, compares push-down stacks—called return stacks—in terms of number of registers (NR) and the bit size of each. Also processor registers are divided into accumulators (ALU), index registers (XR) and remaining registers (GP) for general use.

tape as well as printers.

A number of loaders are available to complete the coding process. With these, which can be stored in ROMs, assembled programs are loaded into read-only memory. They can also be loaded into RAMs, in which case a bootstrap type is used. A so-called relocating loader automatically adjusts program addresses and loads the resulting instructions. And some loaders have linking capability that lets you use routines with undefined labels. These types supply the missing cross-references between separate routines.

Several manufacturers also offer compilers, which allow programs to be written in a high-level language. The benefits are many: A short readable compiler statement corresponds to many symbolic assembly-language statements. Compilers eliminate the need to write detailed codes to control loops, to access complex data structures or to program formulas and functions. And since programming details are lessened, errors are reduced.

But while high-level language programs are compact, easy to read and much easier to write, the net result could be excessive storage space and slower execution, when compared with an assembly-language program. Generally a choice between the two approaches depends on the degree of optimization required and the design time allowable.

In addition to these design aids, test programs —such as simulators—are virtually mandatory to track down the various subtle errors that may remain. Similarly hardware prototype units are essential to the development of the final product. Prototype units generally involve expanded memory capability, teletypewriter or card-reader interface, power supply, chassis and control panel —in addition to a microcomputer.

Besides boosting initial development costs for the designer, the wide range of hardware/software support requires a major investment by the semiconductor manufacturers. This investment is in addition to that needed to produce the LSI chips. In fact, one indication of a vendor's seriousness in marketing a particular microprocessor is the availability of hardware/software support for the product.

Still, ever more manufacturers are entering the field because of the high potential payoff. Various sources predict that the microcomputer market—valued at under $50-million last year—should reach at least $500-million in four years. In the process, a sizable chunk of the TTL market could be replaced. Major microprocessor-chip vendors—such as Intel, National Semiconductor and Rockwell International—are meeting the challenge in a variety of ways.

The Components

By any standard, the recognized leader in microprocessors is Intel, which introduced the product in 1971. Benefiting from its early, one-to-two-year lead over competitors, Intel reportedly captured as much as three-quarters of last year's microcomputer market. Moreover each of its processor chips has been a first of its kind, beginning with a 4-bit unit, the 4004, and leading to an 8-bit PMOS model, the 8008, and the latest advance, an 8-bit NMOS microprocessor—the 8080.

Among the Intel products, the 8080 sets the pace for increased speed and improved instructions. The silicon-gate processor has a 2-μs instruction cycle and 74 basic instructions, which include the 48 instructions of the earlier 8008. The additional 30 instructions and a 6:1 faster execution rate provide up to a 10:1 speed advantage over the 8008. Moreover the improved performance of the 8080 is obtained with a typical power dissipation of only 600 mW, the same as that of the 8008.

The 8080 can address up to 65-k bytes of memory without need for an external address register. This compares with 16-k bytes of memory and an external register for the 8008. The 8080 requires only six peripheral ICs, as contrasted with the 20 needed with the 8008. The NMOS 8080 comes in a 40-pin package and operates from +12 and ±5-V supplies.

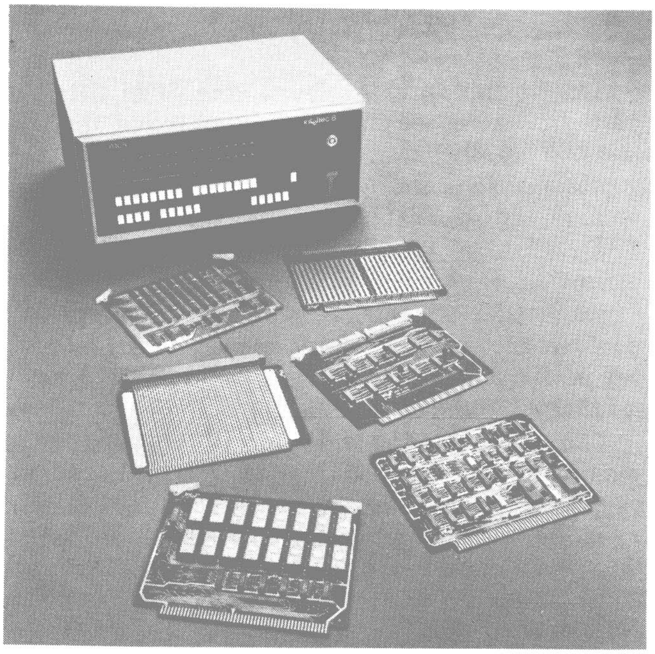
A number of architectural differences account for the improved performance of the 8080. For example, it contains a 16-bit stack pointer (to operate the external LIFO stack) and a 16-bit program counter (to indicate the next instruction), instead of an address storage stack with eight 14-bit locations. A portion of the external memory can be used as a last-in, first-out stack, addressed by the stack pointer upon the execution of a Call, Return or Restart instruction.

Moreover not only the program counter but also the data register, the accumulator and the flags (bits that are set to indicate various conditions) can be saved in the external push-down stack. As a result, multiple interrupts can be handled more easily with the 8080.

The 8080 can perform BCD and binary arithmetic. It also has capability for double-precision arithmetic involving two 16-bit numbers. The NMOS processor can handle up to 256 input ports and a similar range of outputs.

Intel offers several hardware and software design aids for the development of microcomputer systems with its processors. The aids include the Intellec series, consisting of expandable, modular systems that come complete with microprocessor, memory, power supplies and circuitry for teletypewriter interface and clock generation. Each Intellec system is housed in a compact cabinet that features a control and display panel for immediate system monitoring and debugging. All program storage can be accomplished with RAMs, rather than ROMs, for easier program loading and modification. After a program is firm, it can be loaded into ROM.

The standard software package for the Intellec



Most manufacturers offer design aids to simplify development of microcomputer systems. Intel's aids include the Intellec series, which comes complete with memory, supplies and clock and interface circuitry.

series includes a system monitor, contained in pROMs, resident assembler and text editor. A programming module provides the timing and level shifting to program pROMs. Additional support is provided by a cross-assembler and simulator written in Fortran IV and by a PL/M compiler available on time-sharing terminals.

The use of PL/M, derived from IBM's PL/1 language, permits sample programs to be written in a fraction of the time needed to write the same program in assembly language. PL/M offers a far simpler means of programming, compared with assembly language. And the debugging and checkout times of a PL/M program are less, because the structure of the language allows the compiler to detect error conditions that would not be spotted by an assembler.

Recent additions to the Intel product line include several ICs, intended to simplify system design and expansion, and an advanced version of the company's 4-bit microprocessor. The additional circuits include a 4-k-bit dynamic RAM (the 8107), a 2048 × 8-bit ROM (the 8316), a 512 × 8-bit pROM (the 8604), and several peripheral circuits. The latter interface communications lines, handle increased loads and replace IC packages now required.

Intel's new 4-bit microprocessor, the 4014, features software compatibility with the earlier 4004. Additional instructions permit logic operations, such as AND and OR. The 4014 also allows storage capacity of 8-k bytes of memory, increased from the 4-k bytes for the 4004. Other capabilities include improved single-interrupt handling and a single-step mode of operation to simplify testing and debugging.

IMP series—chips, cards and 'boxes'

National Semiconductor offers a broad line of PMOS microprocessor products. They consist of 4, 8 and 16-bit parallel processors that are available in chip form, on card subsystems and in complete microcomputer boxes. Each microprocessor features downward software compatibility—it's compatible with a microprocessor having a shorter word length, but not one with a longer word length. And the fixed instruction set of each can be altered or changed through microprogramming techniques.

The National microprocessors are built around two building-block chips: a Register and Arithmetic Logic Unit (RALU) and a Control Read-Only Memory (CROM). The RALU is a 4-bit "slice"; four are used with one or two CROMs to obtain the 16-bit system, and eight RALUs with two CROMs can be used to form a 32-bit system. The CROM provides storage for the manufacturer's fixed instruction set and the control logic for up to eight RALUs.

The IMP-16C, National Semiconductor's 16-bit microprocessor system, comes on an 8-1/2 × 11-in. PC board. It consists of the processor, clock system, I/O bus drivers, 256 words of RAM and provisions for 512 words of ROM or pROM memory.

The IMP-16C uses a basic 43-instruction set and an expanded 17-instruction set provided by a second CROM. The additional 17 speed processing with instructions that include divide, multiply and double-precision add and subtract. The basic microcycle, or machine cycle, is 1.4 μs. Several microcycles are needed to execute a typical instruction. Two 16-bit numbers can be multiplied for a 32-bit result in a speedy 150 μs.

Each RALU supplies the IMP-16C with an accumulator and a push-down stack. A total of four accumulators improve the bit efficiency of instruction words by cutting down memory-cycle delays. The hardware stack permits rapid nesting of subroutines and interrupts, and its limited depth can be extended—for, say, overflow conditions—by use of main memory.

The microprocessor also features vectored as well as slower, single-line interrupts. With a single-line interrupt, the total overhead time to get to the service routine can be as high as 34.85 μs, since it depends on the number of peripheral devices, and these might number 16. But with a vectored interrupt, the total overhead is only 4.55 μs, a figure that doesn't change with the number of devices.

Several software design aids are offered with each microprocessor. Programs are available for cross and self-assembling, source editing, debugging, and absolute and relocate loading. Also, driver/utility and diagnostic aids are offered. Hardware design aids using assembly language come as complete microcomputers and prototype systems. For the 16-bit system, these are the IMP-16L and 16P boxes. Both units contain a 16-bit microprocessor card.

The IMP-16L has a front-panel display, which provides access to memory and microprocessor registers. The box comes with 4-k, 16-bit words, expandable to 65-k words. Also, a high-speed asynchronous bus permits direct-memory access by peripheral devices without the need to go through the RALUs.

The IMP-16P can interface with a teletypewriter for application software development. This prototyping tool comes complete with chassis, control panel, power supplies and one or more 4-k, 16-bit-word read/write memory modules. National Semiconductor says the IMP-16P is the box to start with to begin a 16-bit design.

Recent additions to National Semiconductor's line are an advanced 4-bit microprocessor system and a microprogramming tool to help designers alter or change the fixed instruction set. The 4-bit system performs BCD arithmetic and uses one CROM, one RALU and a Four-bit Interface Logic Unit, called FILU. The interface unit combines the functions of a number of standard-TTL peripheral circuits.

The microprogramming tool, called Field-Alterable Control Element (FACE), makes use of the fact that a change of CROMs changes the fixed instruction set. FACE, which comes on a card, replaces the CROM on a microprocessor board to form what mainframe designers call writable-control store. Microprocessor control logic now becomes accessible to external ROM or RAM for development of a tailored microprogram. The final result can be stored in a custom-masked CROM or a bipolar pROM. The increased speed of the bipolar memory can reduce delays incurred through connection of external, off-the-board circuitry.

PPS family provides chip sets

Aiming to reduce the number of additional components often needed with single-chip microprocessors, the Microelectronics Div. of Rockwell International offers chip sets for both 4 and 8-bit systems. The sets come complete with processor, clock circuit, memory and input/output ICs. For additional flexibility, you can get a combination ROM and RAM chip or you can select from a growing number of interface and peripheral circuits.

Rockwell estimates that up to 30 peripheral circuits can be eliminated by use of the PPS-4 or PPS-8 chip sets. These are 4 and 8-bit parallel processing systems, respectively. Moreover some of the special circuits in the Rockwell chip sets are not generally available with competing processors. For example, a nonvolatile RAM is being developed for the PPS-4 set. And a controller circuit, in an advanced development stage, for the PPS-8 will permit direct memory access. Rockwell is also developing other circuits for use with peripherals such as CRTs, printers and floppy discs.

The PPS-8 features over 90 instructions, with capabilities for decimal and binary arithmetic single-byte subroutine call and digit/byte manipulation. A complete instruction cycle can be executed in 4 μs, and decimal addition or subtraction is performed at 12 μs per digit. Among 4-bit microprocessors, the PPS-4 sets the pace for speed, with an instruction cycle of 5 μs and a register-to-register add time of 2.5 μs. The 50 instructions for the PPS-4 contain logic and conditional and unconditional data-transfer operations.

The PPS chip sets use a relatively slow clock, typically 200 kHz. However, signals internal to the system are handled at four times that rate. The clock generator provides the processor with two synchronized signals, which are then divided logically into four phases, thus boosting the speed.

Bus lines transfer data during the second and fourth time intervals. In the alternate intervals, address and data-bus lines are automatically cleared to zero. This interface timing scheme permits direct connection of an extensive number of circuits. For the PPS-4, up to 30 circuits can share the bus without need for additional buffering or drive circuitry.

Also, the control logic within the processor allows arithmetic or logic instructions to be carried out in one cycle time. Addition of two decimal digits requires six instructions, or six cycle times. Hence for the 5-μs cycle of the PPS-4, two decimal