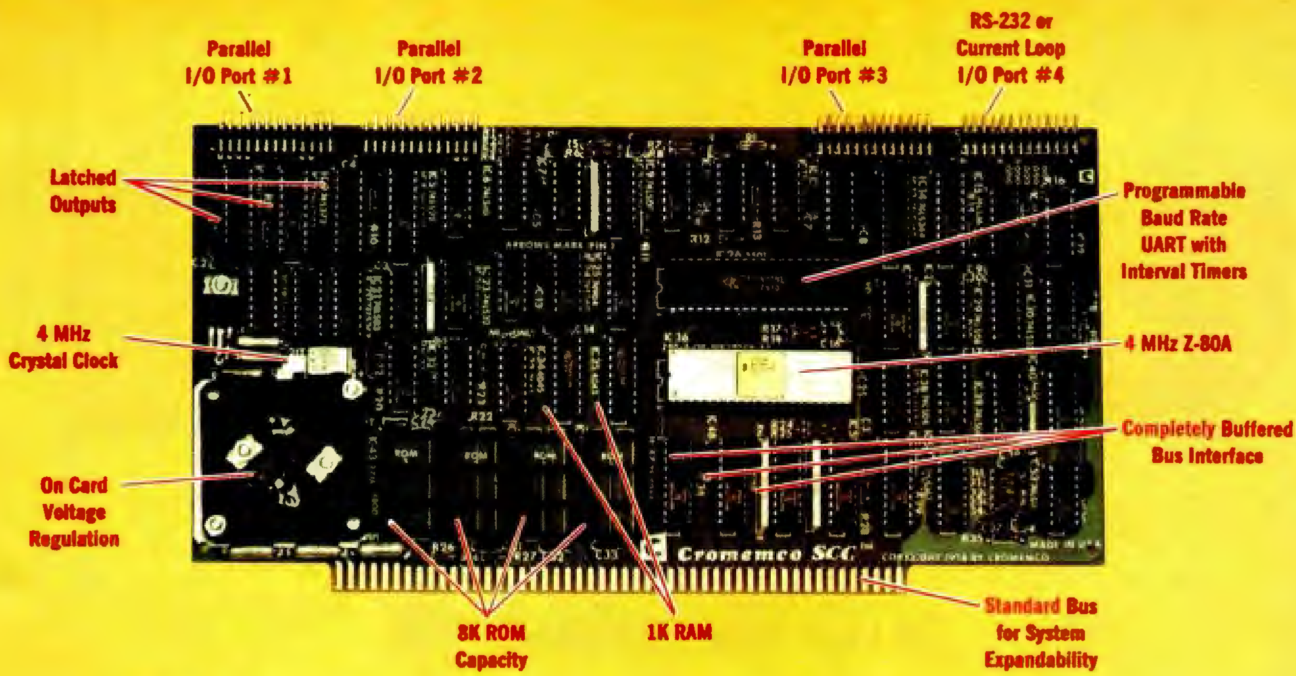
DECEMBER 1978 VOLUME 3, Number 12

BYTE

the small systems journal

$2.00 in USA

$2.40 in CANADA



The single card computer with the features that help you in real life

COMPLETE COMPUTER

In this advanced card you get a professional quality computer that meets today's engineering needs. And it's one that's complete. It lets you be up and running fast. All you need is a power supply and your ROM software.

The computer itself is super. Fast 4 MHz operation. Capacity for 8K bytes of ROM (uses 2716 PROMs which **can be programmed by our new 32K** BYTESAVER® PROM card). There's also 1K of on-board static RAM. Further, you get straightforward interfacing through an RS-232 serial interface with ultra-fast speed of up to 76,800 baud — software programmable.

Other features include 24 bits of bidirectional parallel I/O and five onboard programmable timers.

Add to that vectored interrupts.

ENORMOUS EXPANDABILITY

Besides all these features the Cromemco single card computer gives you enormous expandability if you ever need it. And it's easy to expand. First, you can expand with the new Cromemco 32K BYTESAVER PROM card mentioned above. Then there's Cromemco's broad line of S100-bus-compatible memory and I/O interface cards. Cards with features such as relay interface, analog interface, graphics interface, optoisolator input, and A/D and D/A conversion. RAM and ROM cards, too.

Card Cage

32K BYTESAVER PROM card

EASY TO USE

Another convenience that makes the Model SCC computer easy to use is our Z-80 monitor and 3K Control BASIC (in two ROMs). With this optional software you're ready to go. The monitor gives you 12 commands. The BASIC, with 36 commands/functions, will directly access I/O ports and memory locations — and call machine language subroutines.

Finally, to simplify things to the ultimate, we even have convenient card cages. Rugged card cages. They hold cards firmly. No jiggling out of sockets.

AVAILABLE NOW/LOW PRICE

The Cromemco Model SCC is available now at a low price of only $450 factory assembled ($395 kit).

So act today. Get this high-capability computer working for you right away.

Cromemco
incorporated
Specialists in computers and peripherals
280 BERNARDO AVE., MOUNTAIN VIEW, CA 94040 • (415) 964-7400

Circle 80 on inquiry card.

Model Z-2
Up to 512K of RAM/ROM

Model Z-2D
One or two disks
Up to 512K of RAM/ROM
Up to 184K of disk

System Two
Dual disk
Up to 512K of RAM/ROM
Up to 184K of disk

Fill your computer needs with the industry's most professional microcomputers

#1 IN RELIABILITY

When you choose Cromemco you get not only the industry's finest microcomputers but also the industry's widest microcomputer selection.

What's more, you get a computer from the manufacturer that computer dealers rate #1 in product reliability.*

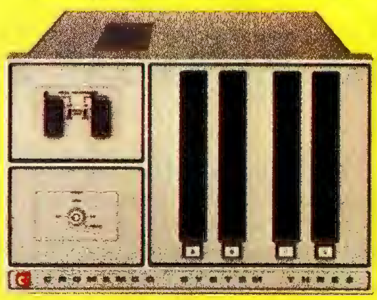
Your range of choice includes our advanced System Three with up to four 8″ disk drives. Or choose from the System Two and Z-2D with 5″ drives. Then for ROM-based work there's the Z2. Each of these computers further offers up to ½ megabyte of RAM (or ROM).

We say these are the industry's most professional microcomputers because they have outstanding features like these:

- **Z-80A microprocessor — operates at 250 nano second cycle time — nearly twice the speed of most others.**

*Rated in ***The 1977 Computer Store Survey*** by Image Resources, Westlake Village, CA.

Up to 512 kilobytes of RAM and 1 megabyte of disk storage



System Three
Two to four disks
Up to 512K of RAM/ROM
Up to 1 megabyte of disk

- **30-amp power supply — more than adequate for your most demanding application.**
- **21 card slots to allow for unparalleled system expansion using industry-standard S-100 cards.**
- **S-100 bus — don't overlook how important this is. It has the industry's widest support and Cromemco has professionally implemented it in a fully-shielded design.**
- **Cromemco card support of more than a dozen circuit cards for process control, business systems, and data acquisition including cards for A-D and D-A conversion, for interfacing daisy-wheel or dot-matrix printers, even a card for programming PROMs.**
- **The industry's most professional software support, including COBOL, FORTRAN IV, 16K Disk-Extended BASIC, Z-80 Macro Assembler, Cromemco Multi-User Operating System, Data Base Management System, Word Processing System — and more coming.**
- **Rugged, professional all-metal construction for rack (or bench or floor cabinet) mounting. Cabinets available.**

FOR TODAY AND TOMORROW

Cromemco computers will meet your needs now and in the future because of their unquestioned technical leadership, professionalism and enormous expandability.

See them today at your dealer. There's no substitute for getting the best.

Cromemco
i n c o r p o r a t e d
Specialists in computers and peripherals
280 BERNARDO AVE., MOUNTAIN VIEW, CA 94040 • (415) 964-7400

Circle 80 on inquiry card.

In the Queue

BYTE December 1978
Volume 3 Number 12

Foreground

Background

Nucleus

BYTE is published monthly by BYTE Publications Inc, 70 Main St, Peterborough NH 03458. Address all mail except subscriptions to above address: phone (603) 924-7217. Address all editorial correspondence to the editor at the above address. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage. Not responsible for lost manuscripts or photos. Opinions expressed by the authors are not necessarily those of BYTE. Address all subscriptions, change of address, Form 3579, and fulfillment complaints to BYTE Subscriptions, PO Box 590, Martinsville NJ 08836. Second class postage paid at Peterborough NH 03458 and at additional mailing offices—USPS Publication No. 102410. Canadian second class registration No. 9321. Subscriptions are $15 for one year, $27 for two years, and $39 for three years in the USA and its possessions. In Canada and Mexico, $17.50 for one year, $32 for two years, and $46.50 for three years. $25 for a one year subscription by surface mail worldwide. Air delivery to selected areas at additional rates available upon request. $25 for a one year subscription by air delivery to Europe. Single copy price is $2.00 in the USA and its possessions, $2.40 in Canada and Mexico, $3.50 in Europe, and $4.00 elsewhere. Foreign subscriptions and sales should be remitted in United States funds. Printed in United States of America.

Subscription WATS Line: (800) 258-5485

In This BYTE

The advent of the personal computer has made possible the calculation of the fast Fourier transform (FFT) on the small system. Applications of this powerful design tool include speech and music analysis as well as circuit design and development. Read **Fast Fourier Transforms on Your Home Computer** by William D Stanley and Steven J Peterson.

page 14

Quite often a software approach to a problem is easier to implement than a hardware approach to the same problem. Tom Munnecke describes the software used in **Designing a Universal Turing Machine** and compares it to a comparable hardware approach.

page 26

Steve Ciarcia describes a simple but useful addition to your computer in **Build an Octal/Hexadecimal Output Display.** This circuit can help you to convert from octal to hexadecimal (and vice versa) or give you the status of a byte during program execution.

page 32

Our cover theme this month (painted by Robert Tinney) is the game of Life. In **Life with Your Computer,** Justin Milliun, Judy Reardon and Peter Smart give a starting point for developing your own version of this exciting game. *page 45*

Researchers probing cellular automata have used Conway's game of Life as a tool in creating a collection of strange and exciting patterns. In David Buckingham's article **Some Facts of Life** we find a description of discoveries made since the original flurry of activity several years ago. *page 54*

One-Dimensional Life is an intriguing variant on John Conway's famous game. Out of this restricted format comes a surprising variety of familiar Life figures such as the flip flop and glider. Dr Jonathan K Millen leads us down the Life line in **One-Dimensional Life.** *page 68*

The same folks who brought you Chess 4.6 now bring you a new, improved version, Chess 4.7. Read the story of the epic battle of the mighty computer and the tenacious, clever human chess master in an article by J R Douglas, **Chess 4.7 versus David Levy.** *page 84*

In many microcomputer applications it is desirable to have a cheap method for printing numerical data. Robert H Astmann describes a way to interface a Texas Instruments 5050M printing calculator to an 8080A based computer in **Interface Your Computer to a Printing Calculator.** *page 94*

When building a computer system it is frequently advisable to have your most often used basic routines stored in read only memory so that they will always be readily available. To make the best use of read only memory, the experimenter should be able to program his own. G H Gable describes one system for programming read only memory in his article, **Zapper: A Computer Driven EROM Programmer.**

page 100

Are you having trouble affording enough hardware to support a high level language such as BASIC? Are you finding it difficult to program in your machine's assembler language? If your answer to either of those questions is "yes," then what you're looking for is **An Easy Programming System.** In this article, Joe Weisbecker gives an introduction to hexadecimal interpretive programming, an alternative to high level languages and assemblers alike. *page 108*

Computer aided instruction is an excellent microcomputer application. To perform this function correctly, it helps to have a programming language designed for the purpose. Prof George A Gerhold describes some of the features such a language should possess in **Teaching with a Microcomputer.** *page 124*

If you have a need for multiplication and division circuits and don't want to worry about timing diagrams, read Mike Weed's discussion of some **Clockless Multiplication and Division Circuits** you can work with. *page 128*

This month we present the second half of **Chess 0.5** in the series **Creating a Chess Player** by Peter W Frey and Larry R Atkin. The program was written by Larry Atkin, who is coauthor with David Slate of the world championship computer chess program, Chess 4.6. The program is written in Pascal and is readily adaptable to personal computers having Pascal systems such as the UCSD Pascal project software. *page 140*

To get the most out of your floppy disk units, you should know how to handle the data that will be stored on them efficiently. A I Halsema introduces us to the concept of **Partitioned Data Sets** and briefly describes a method for implementing them.

page 168

What is the world going to be like in twenty years? That's a difficult question to answer, but the chances are that microcomputers will be part of it. Lawrence Willard takes a lighthearted look at one possible future in his story, **The Mother Chip.** *page 186*

FORTRAN is one of the antecedents to a number of computer languages. The ever popular BASIC is in some respects a simplification of FORTRAN. A number of later languages build upon the computer science learning experience which was FORTRAN and its compilers in the late 1950s and early 1960s. FORTRAN is even now becoming available in floppy disk based systems at the high end of the personal computing performance range. In this issue, W Douglas Maurer provides readers with an article on **FORTRAN and Its Generalizations,** good background reading on an important and still much used language. *page 194*

Editorial

New Wonders of the Computer Age

by Carl Helmers

In recent months, trends in the development of integrated circuit technology have reached new highs of accomplishment, such that it is possible to note some exciting possibilities for design in the next year or so. These new highs are on a broad front of semiconductor technology which is required for the small computers our readers buy and use. The effects of this new technology may not be seen in the retail marketplace for another year or so, since there is a finite design delay time between the availability of a part's design specification and its appearance in finished products.

The first new high in semiconductor technology is the announcement of several new 64 K bit dynamic memory parts (see the Texas Instruments TMS 4164 described on page 209 of this issue). What are the implications of this technology for personal computers? Quite simply, they are new lows in prices for the same functions we see now in the marketplace. Eventually the prices of the 64 K parts will fall to the under $20 level now seen in 16 K chips purchased in volume. Where it once took 32 chips of 16 K bits per chip to saturate a personal computer's address space, it will now take only eight chips (and perhaps a dynamic memory controller chip) to do the same thing. It is now possible to combine a current model microprocessor, a video controller chip, a dynamic memory interface chip, a floppy disk controller chip, and one or two parallel interface chips with eight memory chips and obtain a very complete electronics module for a small computer that uses only 13 or 14 integrated circuits, yet has the performance of a large scale minicomputer of several years ago. In short, the memory address

Continued on page 205

Articles Policy

BYTE is continually seeking quality manuscripts written by individuals who are applying personal computer systems, designing such systems, or who have knowledge which will prove useful to our readers. For a more formal description of procedures and requirements, potential authors should send a large (9 by 12 inch, 30.5 by 22.8 cm), self-addressed envelope, with 28 cents US postage affixed, to BYTE Author's Guide, 70 Main St, Peterborough NH 03458.

Articles which are accepted are purchased with a rate of up to $50 per magazine page, based on technical quality and suitability for BYTE's readership. Each month, the authors of the two leading articles in the reader poll (BYTE's Ongoing Monitor Box or "BOMB") are presented with bonus checks of $100 and $50. Unsolicited materials should be accompanied by full name and address, as well as return postage.■

Sol.® The small computer that won't fence you in.

A lot of semantic nonsense is being tossed around by some of the makers of so-called "personal" computers. To hear them tell it, an investment of a few hundred dollars will give you a computer to run your small business, do financial planning, analyze data in the engineering or scientific lab — and when day is done play games by the hour.

Well, the game part is true. The rest of the claims should be taken with a grain of salt. Only a few personal computers have the capacity to grow and handle meaningful work in a very real sense. And they don't come for peanuts.

Remember, there's no free lunch.

So before you buy any personal computer, consider Sol.® It costs more at the start but less in the end. It can grow with your ability to use it. Sol is not cheap. But it's not a delusion either.

Sol small computers are at the very top of the microcomputer spectrum. They stand up to the capabilities of mini systems costing four times as much.

No wonder we call it the serious solution to the small computer question.

Sol is the small computer system to do the general ledger and the payroll. Solve engineering and scientific problems. Use it for word processing. Program it for computer aided instruction. Use it anywhere you want versatile computer power!

Build computer power with our software.

At Processor Technology we've tailored a group of high-level languages, an assembler and other packages to suit the wide capabilities of our hardware.

Our exclusive Extended BASIC is a fine example. This BASIC features complete matrix functions. It comes on cassette or in a disk version which has random as well as sequential files.

Processor Technology FORTRAN is similar to FORTRAN IV and has a full set of extensions designed for the "stand alone" computer environment.

Our PILOT is an excellent text oriented language for teachers.

Sold and serviced only by the best dealers.

Sol Systems are sold and serviced by an outstanding group of conveniently located computer stores throughout the U.S. and Canada.

For more information contact your nearest dealer in the adjacent list. Or write Department B, Processor Technology, 7100 Johnson Industrial Drive, Pleasanton, CA 94566. Phone (415) 829-2600.

In sum, all small computers are not created equal and Sol users know it to their everlasting satisfaction.

ProcessorTechnology

Why Apple II is the world's best selling personal computer.

Which personal computer will be most enjoyable and rewarding for you? Since we delivered our first Apple® II in April, 1977, more people have chosen our computer than all other personal computers combined. Here are the reasons Apple has become such an overwhelming favorite.

Apple is a fully tested and assembled mainframe computer. You won't need to spend weeks and months in assembly. Just take an Apple home, plug it in, hook up your color TV* and any cassette tape deck—and the fun begins.

To ensure that the fun never stops, and to keep Apple working hard, we've spent the last year expanding the Apple system. There are new peripherals, new software, and the Apple II Basic Programming Manual. And wait till you see the Apple magazine to keep owners on top of what's new.

Apple is so powerful and easy to use that you'll find dozens of applications. There are Apples in major universities, helping teach computer skills. There are Apples in the office, where they're being programmed to control inventories, chart stocks and balance the books. And there are Apples at home, where they can help manage the family budget, control your home's environment, teach arithmetic and foreign languages and, of course, enable you to create hundreds of sound and action video games.

When you buy an Apple II you're investing in the leading edge of technology. Apple was the first computer to come with BASIC in ROM, for example. And the first computer with up to 48K bytes RAM on one board, using advanced, high density 16K devices. We're working to keep Apple the most up-to-date personal computer money can buy. Apple II delivers the features you need to enjoy the real satisfaction a personal computer can bring, today and in the future.

15 colors & hi-resolution graphics, too.

Don't settle for a black and white display! Connect you Apple to a color TV and BASIC gives you instant command of three display modes: Text, 40h x 48v Color-graphics in 15 color and a 280h x 192v High Resolution array that lets you plo graphs and compose 3-D image Apple give you the adde capability of combinin text and graphics, too.

Back to basics, and assembly language too.

Apple speaks three languages: fa integer BASIC, floating point BASIC for scientific and financial applications, and 6502 assembly language. That's maximum programming flexibility. And to preserve user's space, both integer BASIC and monitor are permanently stored in 8K bytes of ROM, so you have an easy-to-use, universal languag instantly available. BASIC gives you graphic commands: COLOR=, VLIN, HLIN, PLOT and SCRN. And direct memory access, with PEEK, POKE and CALL commands.

Software: Ours and yours.

There's a growing selection of preprogrammed software from the Apple Software Bank—Basic Finance, Checkbook, High Resolution Graphics and more. Now there's a User Section in our bank, to make it easy for you to obtain programs developed

other Apple owners. Our Software
ınk is your link to Apple owners all
er the world.

live with
ıe sound
f music.

Apple's ex-
usive built-in
ıeaker delivers
e added dimension of sound to your
ograms. Sound to compose electronic
usic. Sound to liven up games and
lucational programs. Sound, so that
ıy program can "talk" back to you.
hat's an example of Apple's "people
ımpatible" design. Another is its light,
ırable injection-molded case, so you
ın take Apple with you. And the
ofessional quality, typewriter-style
eyboard has n-key rollover, for fast,
rror-free operator interaction.

Apple is the
proven computer.

Apple is a state-of-the-art single
oard computer, with advanced LSI
esign to keep component count to a
ninimum. That makes it more reliable.
f glitches do occur, the fully socketed
ooard and built-in diagnostics sim-
lify troubleshooting. In fact, on our
ssembly line, we use Apples to
est new Apples.

pple II plugs into any standard TV using
n inexpensive modulator (not included).
ı California, call 408/996-1010.

Apple peripherals are smart peripherals.

Watch the far right column of this ad each month for the latest in our growing family of peripherals. We call them "intelligent interfaces." They're smart peripherals, so you can plug them in and run them from BASIC without having to develop custom software. No other personal computer comes close to Apple's expandability. In addition to the built-in video interface, cassette I/O, and four A/D inputs with two continuously variable game paddles, Apple has eight peripheral slots, three TTL inputs and four TTL outputs. Plus a powerful, state-of-the-art switching power supply that can drive all your Apple peripherals.

Available now.

Apple is in stock and ready for delivery at a store near you. Call us for the dealer nearest you. Or, for more details and a copy of our "Consumer's Guide to Personal Computers," call 800/538-9696** or write Apple Computer, Inc., 10260 Bandley Drive, Cupertino, CA 95014.

Circle 14 on inquiry card.

apple computer®

New from Apple.

Valuable new series of software packages for investors

Now private investors can generate their own stock market reports and perform critical investment analysis instantly with Apple II. Just log your Apple II computer on to Dow Jones' central data bank with powerful Apple software: the Dow Jones Series. The first two of these highly practical programs are available now.

With Apple's **Stock Quote Reporter** program, a local telephone call links you to Dow Jones' continuously updated stock quotes for more than 6000 companies listed on six major U.S. exchanges. Current activity for stocks in the investors portfolio is delivered automatically: ask/open, bid/close, high, low and last prices, and volume traded.

Our **Portfolio Evaluator** enables you to analyze current value of your portfolio, and short- and long-term gain/loss for each stock—or for your entire portfolio.

Cost of Apple's Dow Jones service is a one time contract fee of $25, which includes the Stock Quote Reporter program. An additional $3 charge is made for the first three minutes of any transaction and 50¢ per minute thereafter.

To take advantage of Apple's new financial services, Apple II users need only a communications card, a modem and an ordinary telephone. This equipment, the Dow Jones Series, and a broad selection of other Apple software are now in stock at your local Apple dealer.

Circle 15 on inquiry card.

Programming is a snap! I'm halfway through Apple's BASIC manual and already I've programmed my own space wars game.

Those math programs I wrote last week–I just rewrote them using Apple's mini-assembler and got them to run a hundred times faster.

Apple's smart peripherals make expansion easy. Just plug 'em in and they're ready to run. I've already added two disks, a printer and the communications card.

Letters

PASCAL PRAISE

I have just finished absorbing the Pascal articles and editorial in the August 1978 BYTE. If I were a crowd, I'd carry you off on my shoulders, cheering.

The pressure of monthly deadlines seems to have reduced most computerist periodicals to compendia of "How I Did This" and "How to Make That." Recent themed issues of BYTE, though, show exceptional maturity and some solid planning. Your reasoned advocacy of a powerful, common language, with supportive material gathered into one reference issue, ranks as the most important contribution yet.

Pascal appears satisfactory for all our purposes. The concept of p-code provides the mechanism for bringing it to fruition.

Onward, computerists! The milling about is over.

Paul F Doering
56 Elmore Rd
Rochester NY 14618

AUTOMATON TRUMPETER LIVES

On pages 105 and 106 of "Antique Mechanical Computers, Part 2" in August 1978 BYTE there are references to automaton trumpeters and a statement that none survive.

I just returned from Europe and observed at the Deutsches Museum in Munich Germany a life-sized automaton trumpeter. One half (left side) is clothed and the other half is exposed so that the mechanism is visible. Pressed for time, I was not able to find out if it still operates, or who constructed it.

William Harmon
2662 Grand Summit
Torrance CA 90505

COMPUTERS AND ADVERTISING

I am working on both a book and a magazine piece about the use of computers in the field of advertising—particularly in media control, including production scheduling, space acquisition and scheduling for print and broadcast media.

I would be grateful if any BYTE readers with experience in this relatively untapped software area could send me any information they may have, including any programming information which might be helpful.

I am also interested in a program which not only includes production and scheduling control, but carries the whole program right through to a daily alert printout, client billing, and acquisition of advertising space and time.

Maybe I'm asking for too much—but I have a feeling that someone out there may have already worked this out or is at work on it.

In any event, to anyone who would care to send me information I can use in my projected book and article, I would be most grateful.

Larry Ashman
1624 Dole St #1004
Honolulu HI 96822

SIGNETICS 2650: A CORRECTION

I have just finished reading "How to Choose a Microprocessor" by Lou Frenzel, page 124, in July 1978 BYTE. I feel that the advice he gives is excellent; however I also feel compelled to correct an inaccuracy in his section on the Signetics 2650. I own and constantly use a 2650 based microcomputer made by the Central Data Corp. This is available in a basic 1 board configuration complete with on board programmable memory, read only memory, cassette and video input and outout (IO). It is expandable to an S-100 system with floppy disks, 8 K and 12 K BASIC interpreters, assembler/editor, and debugging program. Central Data also publishes a regular newsletter to communicate with the already large number of users of this system.

I have programmed PDP-11s, 8080s, the 2650, and a SC/MP. The 2650 instruction set comes closest to the power of the PDP-11, and I find it a real pleasure to use.

Gordon Brandly
RR 2
Fort Sask
Alberta CANADA T8L 2N8

SOME REFERENCES ON NETWORKING AND PROTOCOLS

Concerning your editorial in July 1978 BYTE, there are two pieces of literature which your readers may want to review. Both concern the Octopus computer network in use at Lawrence Livermore Laboratory in Livermore CA, which is one of the campuses of the University of California.

The first article is in *Datamation*, April 19 1973, pages 58 thru 63. The second article is in *Computer Design*, July 1978, pages 77 thru 86.

The most impressive points of the Octopus network design are simplicity of both software and hardware implementation; suitability for use with standard, inexpensive, byte oriented, asynchronous modem hardware; and easy

Continued on page 158

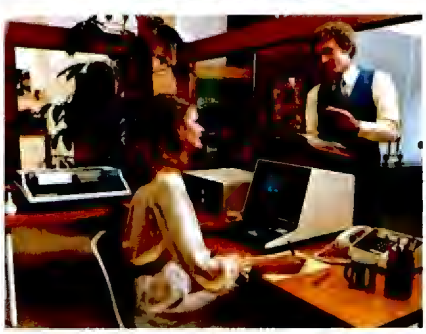
BEFORE YOU BUY COMPUTER #1, VISIT #1 COMPUTERLAND

If the truth is that you want a computer . . . then we want to be your computer store.

We're ComputerLand, the #1 computer store chain in the U.S. What's meaningful about that fact is, that ComputerLand has been chosen by more people as having what they've been looking for. And, since you're looking, let us tell you what you'll find, when you visit a ComputerLand store.

You'll find a product line that's continually evaluated to provide you with the widest and best selection in quality, brand name microcomputers anywhere. You'll find an enthusiastic and knowledgeable staff able to interpret all the equipment specifications, in terms of how they apply to you, and in a way you'll understand. You'll find demonstration areas where you can get a firsthand experience of running a computer yourself.

COMPUTERS FOR BUSINESS



You'll find educational materials to give you a total insight into the world of microcomputers.

You'll find a fully equipped service department to provide whatever assistance is required to keep your computer running in top-notch condition. You'll find computer user's clubs to join, where you can share ideas with people as enthusiastic as yourself. And, with each new visit, you'll find excitement—from the people you deal with, the equipment they offer, and from your own ever-growing personal involvement.

Enough about us. How about what computers do. To attempt to describe all the things your computer might do, would be to describe your imagination. So instead, we'll briefly list some of the many things for which small computers are already being used.

In business, the advent of the versatile and compact microcomputer has put the benefits of computing within reach of small companies. With systems starting at less than $6000, the businessman can computerize things like accounting, inventory control, record keeping, word processing and more. The net result is the reduction of administrative overhead and the improvement of efficiency which allows the business to be managed more effectively.

COMPUTERS FOR THE HOME

In the home, a computer can be used for personal budgeting, tracking the stock market, evaluating investment opportunities, controlling heating to conserve energy, running security alarm systems, automating the garden's watering, storing recipes, designing challenging games, tutoring the children . . . and the list goes on.

In industry, the basic applications are in engineering development, process control, and scientific and analytical work. Users of microcomputers in industry have found them to be reliable, cost-effective tools which provide computing capability to many who would otherwise have to wait for time on a big computer, or work with no computer at all.

COMPUTERS FOR INDUSTRY

And now we come to you, which leads us right back to where we started: **If you want a computer, then we want to be your computer store.**

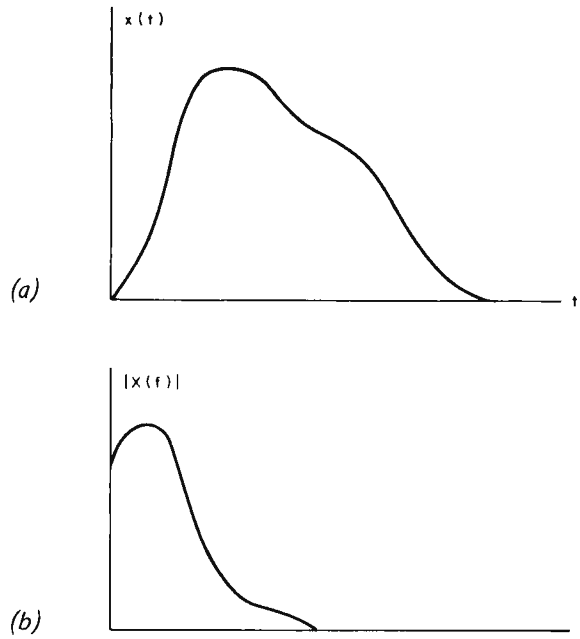
Whether you want a computer for the home, business or industry, come to ComputerLand first. We'll make it easy for you to own your first computer. Because, simply put, we really want your business. When you come right down to it, **that's what makes us #1.**

ComputerLand®

WE KNOW SMALL COMPUTERS

14400 Catalina St., San Leandro, CA 94577 (415) 895-9363 • Franchise Opportunities Available


DYNABYTE DB8/2
DYNABYTE DB8/1
DYNABYTE DB8/4



Fast Fourier Transforms on Your Home Computer

Figure 1: An arbitrary continuous signal x(t) expressed as a function of time (a) may also be described by its spectrum or Fourier transform X(f), which is expressed as a function of frequency (b). The relative strength of the spectrum at different frequencies is a measure of the frequency content that comprises the given signal. The concept of spectrum finds numerous applications in many varied disciplines including music waveform analysis, communications signal analysis, mechanical vibrations, oceanography, statistics, and others. In signal analysis, the function x(t) is said to be a time domain *representation, and X(f) is said to be a* frequency domain *representation.*

William D Stanley
Steven J Peterson
Dept of Electrical Engineering
Old Dominion University
Norfolk VA 23508

The advent of the home computer makes possible many new and varied applications both of a general nature and of a scientific or mathematical nature. One of the latter applications we have successfully implemented on a personal computer is the *fast Fourier transform,* which we will subsequently refer to as the FFT, according to standard usage. Some of the most important properties of the FFT are described in this article, and an FFT program written for the Digital Group Z-80 System using BASIC is provided.

Continuous Fourier Transform

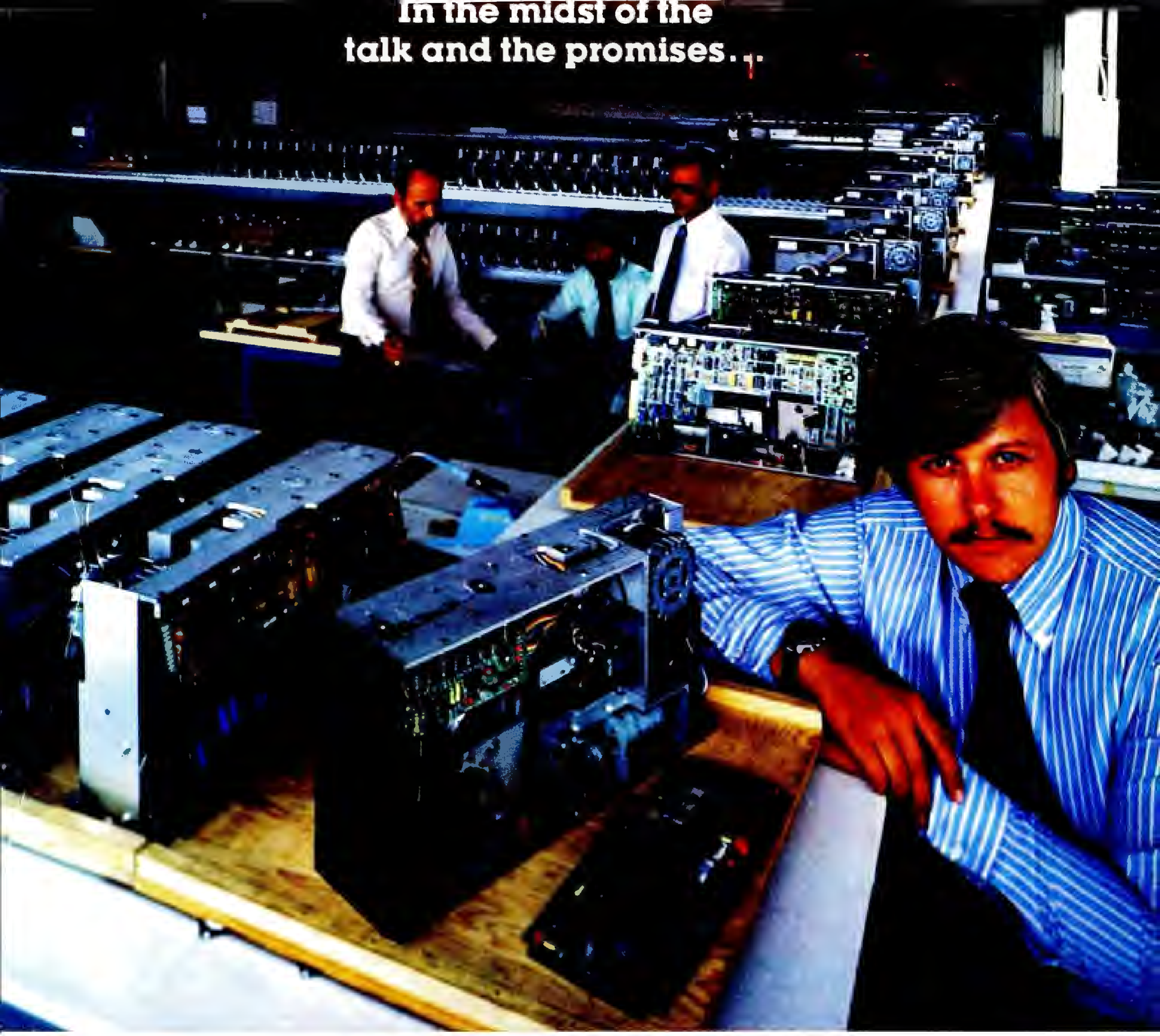
Before discussing the FFT in particular, it is desirable to briefly survey some of the general concepts of the classical continuous Fourier (pronounced "foor-yay") transform. The terminology used refers to *time* and *frequency* since they are among the most common variables of interest in many applications, although the theory involved applies to a variety of different types of physical phenomena.

Consider the waveform x(t) shown in figure 1a which is displayed as a function of *time* (denoted by t). The waveform can also be described by the *frequencies* present in the signal. This description is called the *spectrum* of the time signal and, mathematically, it is the *Fourier transform* of the time function. The process of Fourier transformation is represented by the mathematical function

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\,dt$$

where X(f) is the Fourier transform of x(t). *[The constant j is used in electrical engineering to denote* $\sqrt{-1}$, *also called i. The number e, 2.71828, is the base of the natural algorithms. . . .CM]* For all but fairly simple functions, this mathematical process represented a formidable operation for many years. Prior to the development of the digital computer, many analytical and experimental methods were investigated for determining the approximate spectra of functions that arose in physical systems.

The magnitude of a typical spectrum is shown in figure 1b and is denoted by |X(f)|, where f represents the frequency in Hertz (Hz). For example, if x(t) were a music signal, strong peaks of the spectrum at low frequencies would be characteristic of a significant amount of bass content such as



PerSci delivers the Dual-Headed Diskette Drive that works and the Double Density IBM Compatible Controller to back it up.

While everyone else was waiting and promising—PerSci did it! Designed the industry's first fully reliable, field proven, dual-head design—now available in a high technology dual diskette drive, the PerSci Model 299. The 299 has a total of four heads handling two double-sided diskettes—providing up to 3.2 Mbytes of data in a drive the size of a standard floppy. And because the Model 299 has voice coil positioning, this dual diskette drive can seek data four to six times faster than possible on two independent stepper motor drives. All offered by PerSci at a highly competitive OEM price.

To save time and money in systems design, PerSci backs up the 299 with the first available double density, dual-head flexible disk controller—the Model 1170—for all applications requiring IBM 2D, IBM 3740 or S-100 bus compatibility. The PerSci 1170 is Z-80 based, the smallest, most intelligent floppy disk controller on the market.

The PerSci 299 and 1170 controller are designed with the same advanced technology that has made PerSci number three in diskette product shipments (and moving up). Don't wait around for more talk and more promises. Call PerSci for full information.

PerSci, Inc., 12210 Nebraska Avenue,
West Los Angeles, California 90025.
(213) 820-3764.

PERSCI

Peripherals a Generation Ahead.

Circle 289 on inquiry card.

drums or tubas. Conversely, many string instruments such as the violin display stronger peaks at higher frequencies in the audio spectrum. The frequency spectrum (or Fourier transform) thus provides a plot of the relative weight of different frequencies that comprise or represent the given signal.

If the Fourier transform or spectrum of a signal is known, the time function may be determined from the inverse transformation which is given by

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}\,df$$

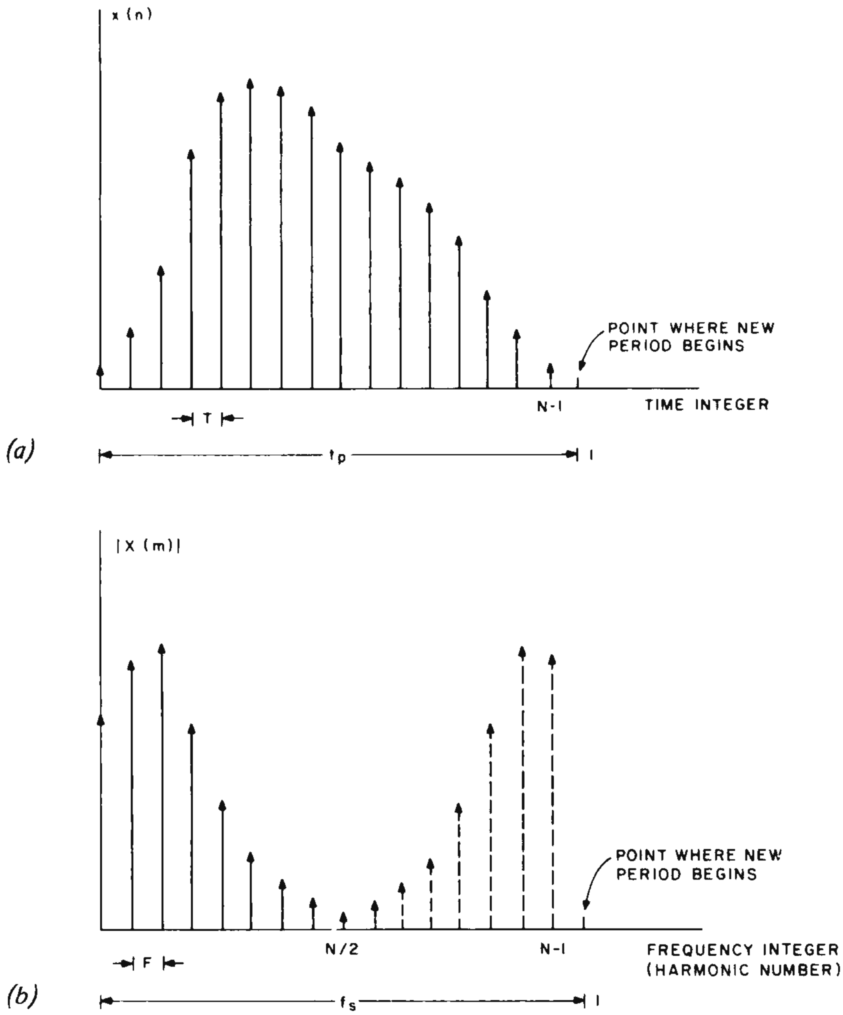


Figure 2: A sampled function of time (a) and its discrete Fourier transform spectrum (b). The discrete Fourier transform (DFT) functions are used to approximate the continuous transform functions whenever digital implementation is to be used. The time function is sampled at N points separated by an increment T over an interval t_p = NT to create a discrete function x(n). The resulting spectrum X(m) is periodic with a period $f_s = 1/T$ and contains N components within one period with spacing between components $F = 1/t_p$. If x(n) is a real function, only half or N/2 of the spectral components are unique. The integers n and m represent the time and frequency integers which identify the location in the sequence of the time sample (t = nT) and the frequency component or harmonic number (f = mF).

Observe that the inverse transform has essentially the same general form as the direct transform except for the sign of the exponential argument.

The concept of the frequency spectrum has long played a most important role in numerous scientific applications and has been of interest to mathematicians, engineers and scientists of many different disciplines. Among the areas where spectral analysis has been employed are sound and music analysis, communications systems design, analysis of mechanical vibrations, ocean wave analysis, statistics and many others.

Discrete Fourier Transform

The heart of the FFT is a mathematical operation known as the *discrete Fourier transform* (DFT). In the DFT, a set of integers n and m are defined to represent the equivalent in a sense of the time and frequency variables, respectively, of the continuous Fourier transform. This correspondence is best seen by observing the sampled signal x(n) shown in figure 2a. There are assumed to be N samples of the signal spaced T seconds apart. Thus, as n varies from 0 to N−1, the N samples of the time signal are generated. The duration of the time signal is t_p = NT.

The DFT of x(n) is defined by the finite summation

$$X(m) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)W^{mn}$$

where

$$W = e^{\frac{-j2\pi}{N}}$$

The function X(m) represents a discrete spectrum with m serving the same purpose in frequency as n did in time. The frequency increment between successive components is $F = 1/t_p$ so that the spectral component at a frequency mF is X(m). For x(n) real and for N time points, a unique spectrum can be computed only at N/2 frequency points. Actually, X(m) is periodic in m with N points in each period, but only N/2 are unique. X(m) is, in general, a complex function consisting of a real and an imaginary part at each frequency. For many applications, the magnitude spectrum |X(m)| is the quantity of most significance. Some of the preceding points are illustrated in figure 2b.

As in the case of continuous signals, an *inverse discrete Fourier transform* (IDFT) can be defined. In this case, the inverse transformation is

It may be a hobby, or it may be an asset... It SHOULD be a Heathkit® Computer System

No matter what your computer system needs may be, Heathkit computers make sense! Heathkit "total design' computer systems give you a wide selection of peripherals, software programs to get you up and running fast; plus the reliability, service and responsibility that come from being a leader in the electronics industry for some 50 years!

OUR 8-BIT COMPUTER

Every Heathkit Computer Product is designed to offer substantial benefits over competitive products on the market. Our 8080A-based H8 for example, is more than just a simple 8-bit machine. With its "intelligent" front panel and keyboard entry and digital display, it actually lets you compute and program without the addition of any peripherals. It's an ideal computer training system, and when you're ready to advance, it's ready too. It's one of the most expandable computers around, and now with its NEW floppy disk system, it could be the only computer you'll ever need.

OUR 16-BIT COMPUTER

If you need the power, speed and versatility of a 16-bit machine, there's nothing better than our H11A. Based on the famous DEC LSI-11/2, the H11A provides complete DEC compatibility and access to the thousands of practical software programs and applications that entails. Along with our own complete systems software and our line of DEC-compatible peripherals including the DEC Writer II and our new floppy disk, you'll have state-of-the-art computing power at its very best!

OUR PERIPHERALS

The Heathkit Computer peripherals offer the same competitive advantages of our two computers. Our H9 CRT terminal, H10 paper tape reader/punch, ECP-3801 cassette storage recorder/player, and our new WH14 line printer, plus the new floppy disk storage systems all give you the quality, performance and value that Heath company is famous for. And we sell the memory, I/O interfaces and accessories you need to custom design a system to your particular specifications!

ALL THIS, PLUS HEATH DOCUMENTATION, SERVICE AND SUPPORT

One of the most important parts of ANY computer system is documentation. And Heath documentation is quite simply, the best around. If you buy our computer products in kit form, you get a comprehensive step-by-step assembly manual that takes you every step of the way from unpacking to final plug-in. The knowledge you gain in building your Heathkit computer is invaluable–for service if it's ever needed, for quick troubleshooting and correction, and just for understanding the workings of the machine. In both our kit and fully assembled products, our comprehensive operating and instruction manuals are fully detailed, thorough and accurate. This documentation, plus Heathkit technical consultants and service nationwide, make your Heathkit computer system one you can depend on–to work right the first time, and to last for years!

SEND FOR YOUR FREE CATALOG

Read all about these Heathkit computer products and peripherals in the new FREE Heathkit catalog. Write Heath Company, Dept. 334-482 for your FREE copy. Or visit your Heathkit Electronic Center (Units of Schlumberger Products Corporation).

HEATHKIT COMPUTERS

Systems Engineered for Personal Computing

Heath Co. Dept. 334-482
Benton Harbor, MI 49022
CP-159

Circle 160 on inquiry card.

$$x(n) = \sum_{m=0}^{N-1} X(m)W^{-mn}$$

The resulting function is periodic in the variable n and has N points in one period. Thus, even if the original time signal were not

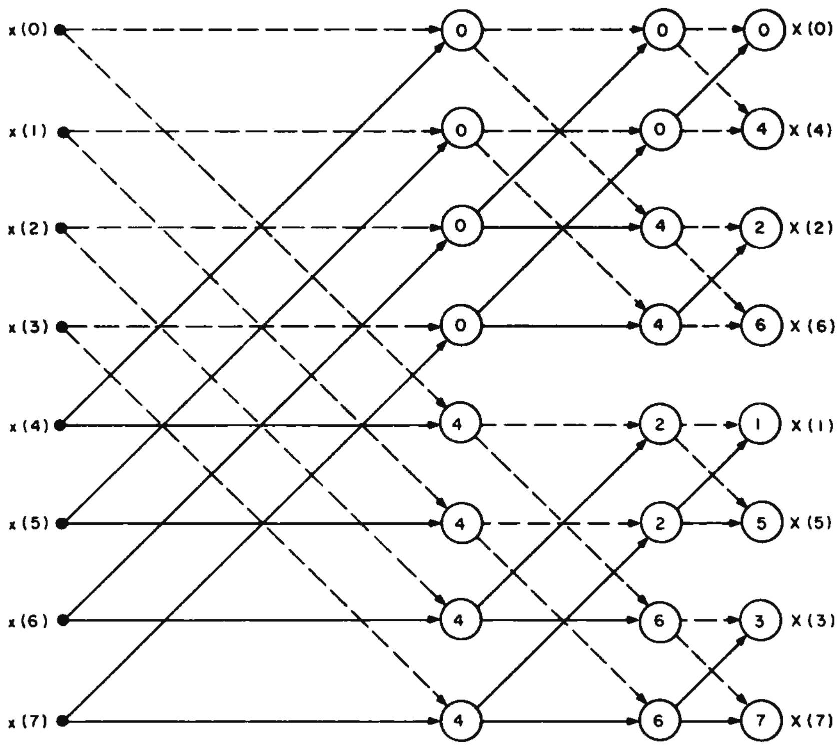


Figure 3: Flow diagram indicating the computations involved in an 8 point fast Fourier transform (FFT) implementation of the discrete Fourier transform (DFT) function. Significant reductions in computation time are achieved with FFT realizations of large arrays. For example, the computation time for a 1024 (2^{10}) sequence of samples using an FFT is approximately 1 percent of the time required by direct application of the DFT. In the chart, two paths merging together in a given column represent a combination of two quantities in the preceding column. For example, the first quantity in the second column is obtained by forming x(0) + W^0x(4). The first term is indicated by the dashed line and the second is indicated by the solid line. The integer in the circle is the power of W. (See text for definition of W.) The pattern continues until the spectrum appears in the last column. This particular algorithm for the FFT results in a scrambled order for the spectral coefficients as can be seen from the chart. Some variations result in a natural order but require more internal memory.

periodic, the operation of the IDFT produces a function capable of providing the desired results in one cycle, but the pattern continues to repeat itself if the interval is extended outside of the basic range.

Observation of the definition of the DFT reveals that there are approximately N complex multiplications and about the same number of complex additions required to compute the spectrum at one particular value of m. Since there are N/2 unique spectral components, the total number of computations required to generate a complete spectrum is of the order of N^2. The Cooley-Tukey algorithm, published in 1965, demonstrates one way to perform this transformation with a number of computations of the order of N $\log_2$(N), which turns out to be an enormous savings in computational time for long signal records. The Cooley-Tukey algorithm, along with subsequent variations, is referred to as the *fast Fourier transform* (FFT). Thus, the FFT is a high speed algorithm for computing the discrete Fourier transform.

While the DFT is a finite summation and the classical Fourier transform is an integral transform, the DFT may be used to closely approximate the continuous function under many circumstances. Some of the concepts involved with such an approximation are considered later in this article.

The various FFT algorithms work best when the number of points in the sample record is an integer power of 2, ie: $N = 2^k$, where k is an integer. The form of one of the basic algorithms is shown in figure 3 for the case of N = 8. Obviously, N = 8 is far too small for most applications, but the flow graph is of interest in understanding the form of the general computational algorithm. This particular algorithm is referred to as an *in place* algorithm since at each stage of the computation, the data may be stored in the same memory locations from which they were obtained.

Implementation of In Place Algorithm

The in place algorithm previously discussed was implemented on the Digital Group Z-80 System using BASIC. The program is given in listing 1. The particular system used had 18 K bytes of memory, of which about 12 K bytes were required for the BASIC software. It was determined that a 256 point transform could be computed with this system and the program listed uses this capacity. It could be readily expanded or contracted as the available memory size dictates. However, the size selected should be chosen as an integer power of 2 as previously noted. Thus, the next smaller size should be 128 and the next larger size should be 512.

In order to reduce the memory requirements, the trigonometric functions are generated as they are required in the program. This approach is not nearly as efficient from the standpoint of computation time as would be the process of initially generating and storing the functions in

Listing 1: Fast Fourier transform routine described in text. Lines 10 to 499 are available for the user to describe the time function that is to be studied.

```
2 DIM X1(256),X2(256)
4 N=256 : L=8 : P1=3.14159
6 REM -- GENERATE TIME FUNCTION --
10 REM
20 REM  LINE NUMBERS 10-499 ARE USED TO
30 REM  GENERATE OR INPUT THE TIME FUNCTION
40 REM
500 PRINT "DO YOU WANT A LISTING OF THE GENERATED TIME FUNCTION ";
510 INPUT A$
520 IF A$="NO" THEN 640
530 IF A$<>"YES" THEN 500
540 B=X1(0)
550 FOR Z=0 TO N-1
560 IF ABS(X1(Z))>B THEN B=ABS(X1(Z))
580 NEXT Z
600 FOR Z=0 TO N-1
610 PRINT X1(Z);TAB(41+20*X1(Z)/B);"*"
620 NEXT Z
630 REM  - SCALE INPUT TIME FUNCTION -
640 FOR Z=0 TO N-1
650 X1(Z)=X1(Z)/N
660 NEXT Z
670 REM - - FFT IN-PLACE ALGORITHM - -
675 PRINT" - FFT CALCULATION IN PROGRESS -"
680 I1=N/2 : I2=1 : V=2*P1/N
690 FOR I=1 TO L
700 I3=0 : I4=I1
710 FOR K=1 TO I2
720 X=INT(I3/I1)
730 GOSUB 1300
740 I5=Y
750 Z1=COS(V*I5)
760 Z2=-SIN(V*I5)
770 FOR M=I3 TO I4-1
780 A1=X1(M) : A2=X2(M)
790 B1=Z1*X1(M+I1)-Z2*X2(M+I1)
800 B2=Z2*X1(M+I1)+Z1*X2(M+I1)
810 X1(M)=A1+B1 : X2(M)=A2+B2
820 X1(M+I1)=A1-B1 : X2(M+I1)=A2-B2
830 NEXT M
840 I3=I3+2*I1 : I4=I4+2*I1
850 NEXT K
860 I1=I1/2 : I2=2*I2
870 NEXT I
880 REM - OUTPUT RESULTS -
890 PRINT"IN WHAT FORM DO YOU WANT THE OUTPUT ?"
900 PRINT"   MAGNITUDE SPECTRUM PLOT     (1)"
910 PRINT"   TABLE OF VALUES             (2)"
920 INPUT A
930 IF A=1 THEN 970
940 IF A=2 THEN 1130
950 PRINT"INCORRECT INPUT (1 OR 2)" : GOTO 890
960 REM - OUTPUT MAGNITUDE SPECTRUM PLOT -
970 B=0
975 PRINT" - CALCULATIONS IN PROGRESS -"
980 FOR Z=0 TO N/2
985 X=Z
990 GOSUB 1390
1000 IF X3>B THEN B=X3
1010 NEXT Z
1020 FOR Z=0 TO N/2
1025 X=Z
1030 GOSUB 1390
1040 X4=INT(55*X3/B)
1050 C=0
1060 PRINT Z;TAB(5);"!";
1070 C=C+1
1080 IF C<X4 THEN PRINT"="; : GOTO 1070
1090 PRINT ""
1100 NEXT Z
1110 GOTO 1240
1120 REM - OUTPUT TABLE OF VALUES -
1130 U=0
1140 Z=0
1150 PRINT"HARMONIC";TAB(14);"REAL";TAB(30);
1160 PRINT"IMAGINARY";TAB(50);"MAGNITUDE"
1165 X=U
1170 GOSUB 1390
1180 PRINT U;TAB(10);X1(Y);TAB(30);X2(Y);TAB(50);X3
1190 U=U+1 : Z=Z+1
1200 IF Z>9 THEN 1140
1210 IF U>N/2 THEN 1240
1220 GOTO 1165
1230 REM - TERMINATE ? -
1240 PRINT"DO YOU WANT ANOTHER OUTPUT (YES, NO)";
1250 INPUT A$
1260 IF A$="YES" THEN 890
1270 IF A$<>"NO" THEN 1240
1280 END
1290 REM - SCRAMBLER SUBROUTINE -
1300 Y=0 : N1=N
1310 FOR W=1 TO L
1320 N1=N1/2
1330 IF X<N1 THEN 1360
1340 Y=Y+2^(W-1)
1350 X=X-N1
1360 NEXT W
1370 RETURN
1380 REM - MAGNITUDE (X3) SUBROUTINE
1390 GOSUB 1300
1400 X3=SQRT(X1(Y)^2 + X2(Y)^2)
1410 RETURN
1420 END
```

memory so that they can simply be called as required. However, where speed is not a major priority, this approach minimizes the total memory required.

Statements 10 through 499 in the program represent the particular input signal for which the transform is being computed. The time function may be generated by appropriate equations or an algorithm as will be demonstrated for several cases later. For experimental data, the values could be listed point by point if the function cannot be readily described by an equation.

Applying the Program

In order to effectively utilize an FFT program for spectral analysis, it is necessary to understand some of the peculiarities of the DFT and its relationship to the continuous Fourier transform. Although the time signal may or may not be periodic in nature, the mathematical form of the DFT treats the signal as if it were periodic. The total duration of the time signal is the period t_p, and for the program being considered, this period contains 256 points. If T is the time increment between samples, then $t_p = 256\,T$. The spectrum obtained from the DFT is also periodic and contains N (or 256) spectral components. However, for a time function that is real (which incidentally is the case for all signals considered in this article), it can be shown that half of the components are ambiguous; ie: they are similar to the other half and do not represent any actual spectral information. Thus, there are N/2 (or 128) meaningful complex spectral components that are obtained with the FFT. These components are spaced apart in frequency by $F = 1/t_p$. The value for $m = 0$ corresponds to the DC component, $m = 1$ is the fundamental, $m = 2$ is the second harmonic, etc. According to sampling theory, a time signal must be sampled at a rate at least equal to (practically speaking, greater than) twice the highest frequency contained in the spectrum. Thus, if the highest frequency contained in a spectrum

Identical twins...almost.

**This one prints.
$395.**

**This one prints & plots.
$795.**

And they plug into Apple, Exidy, PET, Radio Shack & most major computers.

In 1977, AXIOM pioneered low-cost electrosensitive line printing, setting an example which others have been quick to follow.

However, we believe that once you are the market leader, you should stay out in front. So we're proud to announce the birth of two exceptional new products, the EX-801 MicroPrinter and the EX-820 MicroPlotter which set *new* standards for versatile low-cost hardcopy.

Meet the MicroPrinter

Here's the answer to a micro (or mini) computer's fondest dreams. Designed around the Intel 8048 microprocessor, the EX-801 MicroPrinter operates to 160 cps (that's 14 times faster than a TTY), and gives you the choice of 3 intermixable character sizes to provide 80, 40 or 20 columns on 5-inch wide electrosensitive paper, making this printer ideal for CRT hardcopy, data logging, remote message printing, program listing, record keeping... In fact, any application needing fast, low-cost copy.

Introducing the MicroPlotter

Our EX-820 MicroPlotter does everything the EX-801 does — plus it plots. Under software control, you have unlimited flexibility to mix alphanumeric ASCII and graphics on any line.

Prices

- EX-801P (Parallel ASCII input) —$395.
- EX-801S (RS232C/20mA to 1200 bps) — $495.
- EX-801HS (RS232C/20mA to 9600 bps) — $549.
- EX-820 (Parallel ASCII and RS232C/20mA) — $795.

Compatibility

At Axiom, we don't see much point in shipping a printer or plotter which requires days of labor, extra connectors, cables and interface cards before it can be made to work with your computer.

So *we* provide the interfaces. Just add the appropriate interface to your choice of printer or plotter for instant compatibility.

Interfaces

For the EX-801P MicroPrinter

- Apple II interface add $100.
- TRS-80 (level II) interface add $100.
- PET interface add $100.
- Exidy interface add $50.

For the EX-820 MicroPlotter (Provides full printing capability in addition to graphics)

- Apple II (HIRES at a single keystroke) add $100.
- PET (All PET graphics) add $100.

These prices are for interfaces purchased with an Axiom printer or plotter. Phone or write today.

AXIOM

AXIOM CORPORATION

5932 San Fernando Rd., Glendale, CA 91202
(213) 245-9244 • TWX: 910-497-2283

Circle 24 on inquiry card.

```
10 REM - GENERATE 25% PULSE
20 FOR Z=0 TO N/4
30 X1(Z)=1
40 NEXT Z
50 FOR Z=N/4 TO N
60 X1(Z)=0
70 NEXT Z
```

```
10 REM - GENERATE 12.5% PULSE
20 FOR Z=0 TO N/8
30 X1(Z)=1
40 NEXT Z
50 FOR Z=N/8 TO N
60 X1(Z)=0
70 NEXT Z
```

```
10 REM - GENERATE 1000HZ SINE WAVE
20 T=0
30 FOR Z=0 TO N-1
40 X1(Z)=SIN(2*3.14159*1000*T)
50 T=T+1.953125E-4
60 NEXT Z
```

```
10 REM - GENERATE 1010HZ SINE WAVE
20 T=0
30 FOR Z=0 TO N-1
40 X1(Z)=SIN(2*3.14159*1010*T)
50 T=T+1.953125E-4
60 NEXT Z
```

Listing 2: Three different generating routines that can be used with listing 1 as the time functions. The first routine generates a pulse function that lasts 25 percent of the time that is being analyzed. The second routine also generates a pulse but half as long as the first routine. The third and fourth routines generate sine waves which are only slightly different.

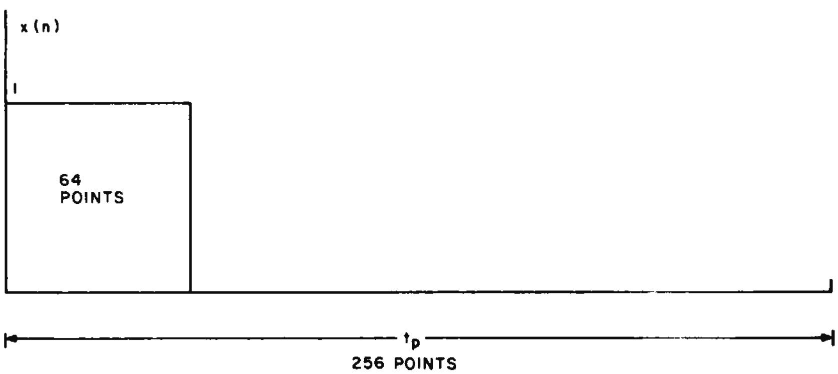
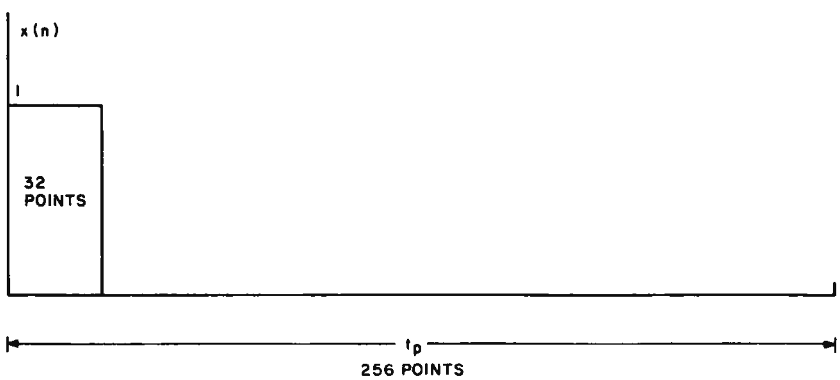


Figure 4: Rectangular pulse for which the FFT is partially displayed in photos 1 and 2. The pulse is unity for 64 of the 256 points in the time record and zero for the remainder.

Figure 5: Rectangular pulse for which the FFT is partially displayed in photo 3. The pulse is unity for 32 of the 256 points in the time record and zero for the remainder. Since this pulse is shorter than the one of figure 4, the spectrum is broader. In general, there is an inverse relationship between the width of a pulse-like time function and the width of the frequency spectrum. This property is an important concept in signal transmission and results in the requirement of larger bandwidths for transmitting shorter pulse signals.



is known to be no greater than f_h, the maximum time between samples (T) should be chosen to satisfy $T < \frac{1}{2f_h}$. If this condition is not met, there will be a spectral overlap or *aliasing* effect which will distort the spectrum.

For a fixed number of points (such as 256 for the program under discussion), there is a trade-off between the high frequency capability and the spectral resolution. In order to analyze higher frequencies, a shorter sampling time is required, but this necessitates a shorter overall period and a larger increment between successive frequencies. Specifically for 256 points, $N/2 = 128$; and since $N = 0$ corresponds to DC, the highest frequency that can be measured is 127 times the spectral resolution. It is very important that the sampling rate be chosen to be greater than twice the highest frequency in the spectrum even if the higher frequencies are not of interest. If the minimum sampling rate requirement is not met, erroneous spectral components may appear at various places in the spectrum.

There are various other properties of the DFT that may be important in applying an FFT program in various situations. The reader is encouraged to consult one of the references listed at the end of this article or the many other available sources for more extensive details, since this article provides only a brief overview of the theory along with the details of a workable program for a home computer.

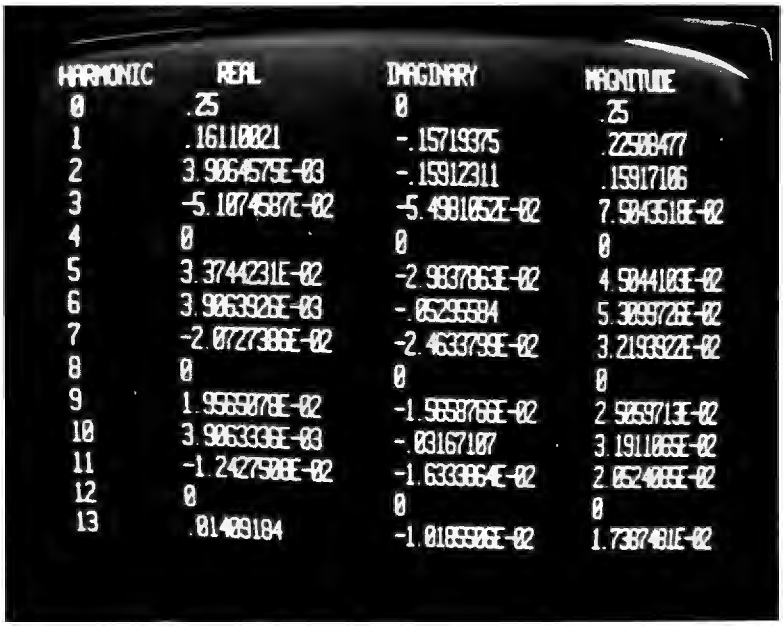
Examples

Several examples that illustrate some of the properties of the FFT are now considered. The various function programs for these waveforms are shown in listing 2. The first example is that of a single rectangular pulse whose duration is 25 percent of the total period corresponding to 256/4, or 64 points as illustrated in figure 4. (Due to the

large number of points, the function is shown as a continuous curve.) The video display of the first 14 spectral components in tabular form is shown in photo 1, and the first 15 components of the magnitude spectrum are displayed in photo 2. Henceforth, only the magnitude spectra will be shown.

When the pulse duration is changed to 12.5 percent of the period or 32 points as indicated in figure 5, the magnitude spectrum changes to the form shown in photo 3.

It should be pointed out that the bandwidth of a rectangular pulse is theoretically infinite in extent and so there is some aliasing error in each of these cases. However, the effects of aliasing are not pronounced in these two examples over the frequency range shown in the photos. At larger harmonic



HARMONIC	REAL	IMAGINARY	MAGNITUDE
0	.25	0	.25
1	.16110021	-.15719375	.22508477
2	3.9064579E-03	-.15912311	.15917106
3	-5.1074587E-02	-5.4981052E-02	7.5043518E-02
4	0	0	0
5	3.3744231E-02	-2.9837863E-02	4.5044103E-02
6	3.9063928E-03	-.05295584	5.3098728E-02
7	-2.0727386E-02	-2.4633799E-02	3.2193922E-02
8	0	0	0
9	1.9569078E-02	-1.5658766E-02	2.5059713E-02
10	3.9063336E-03	-.03167107	3.1911065E-02
11	-1.2427528E-02	-1.6330864E-02	2.0524085E-02
12	0	0	0
13	.01409184	-1.0185926E-02	1.7387481E-02

Photo 1: The first 14 components (DC and harmonics up through the 13th) of the FFT spectrum corresponding to the pulse shown in figure 4. The program lists the real part of X(m), the imaginary part of X(m) and the magnitude |X(m)|.

Photo 2: Video graphics display of the magnitude spectrum corresponding to the pulse shown in figure 4. The display is of course rotated 90° from the basic mathematical form illustrated in figure 2.

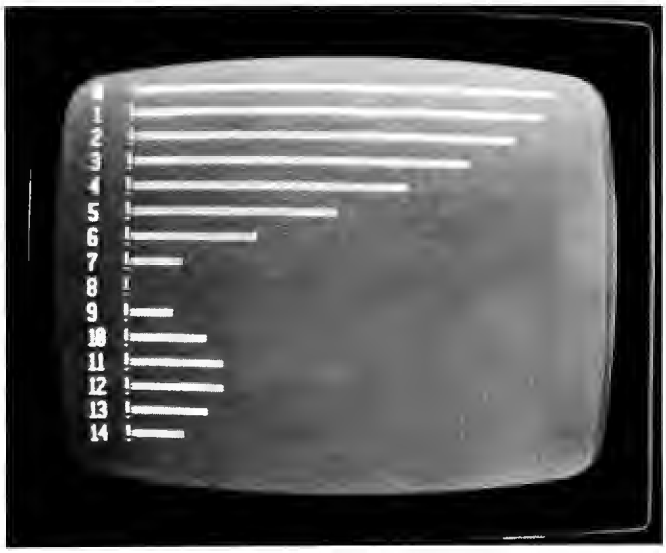


Photo 3: Video graphics display of the magnitude spectrum corresponding to the pulse shown in figure 5.

values for the given signals and at shorter pulse widths for the given frequency range, the aliasing errors would be more significant.

A sine wave representing an assumed frequency of 1000 Hz and an assumed sampling time of T = 0.1953 ms was generated and analyzed. The resulting spectrum is shown in photo 4. Note that the frequency resolution is $F = 1/(0.1953 \times 10^{-3} \times 256) = 20$ Hz so that 1000 Hz corresponds to harmonic number 50. Observe that an ideal single line appears as one might hope. On the other hand, when the frequency is changed to 1010 Hz while maintaining the same value of T, the spectrum changes to the form shown in photo 5. The reasons for the striking difference are as follows: In the first case, the frequency corresponds exactly to one of the harmonic numbers (50th harmonic), and a property of the DFT is that no other line components appear in this case. However, in the second case, the component would theoretically appear halfway between the 50th and 51st harmonics so that the imperfections of the finite time duration of the observed sinusoid are now apparent. The phenomenon observed is called *leakage.* It can also be readily verified that the first sinusoid was observed over an exact integer number of cycles, while in the second case, the sinusoid was truncated during a cycle.

This example illustrates the necessity of understanding some of the limitations of the truncation and sampling processes in order to properly evaluate results. The phenomena just noted can be reduced by smoothing the data to be transformed with certain *window functions* before computing the FFT. Window functions smooth the beginning and end of a record length and reduce the effects of leakage on the spectrum.

More Examples

Other applications include the use of an analog to digital converter to sample speech and music waveforms or the waveforms encountered in electronic systems. The sample points could be stored for later spectral analysis using the FFT program. We hope readers will be encouraged to experiment with the program on their own computers.■

REFERENCES

1. Bergland, G D, "A Guided Tour of the Fast Fourier Transform," *IEEE Spectrum,* July 1969, pages 41 thru 52.
2. Brigham, E O, *The Fast Fourier Transform,* Prentice-Hall, Englewood Cliffs NJ, 1975.
3. Cooley, J W, and Tukey, J W, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Math of Computation,* volume 19, April 1965, pages 297 thru 301.
4. Stanley, W D, *Digital Signal Processing,* Reston Publishing Co (a division of Prentice-Hall), Englewood Cliffs NJ, 1975.

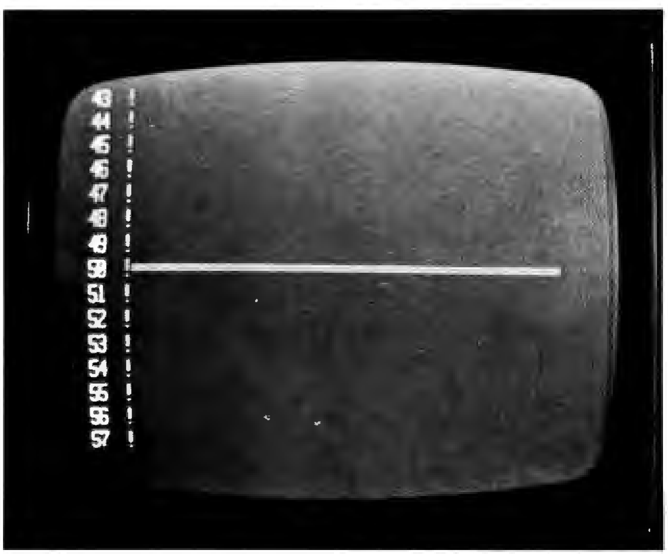


Photo 4: Video graphics display of the magnitude spectrum corresponding to a sine wave whose assumed frequency is 1000 Hz with a sampling interval T = 0.1953 ms. This assumption results in an integer number of cycles (50) in the record duration t_p, which corresponds to 50 ms. The frequency then corresponds exactly to the 50th harmonic and the spectrum appears as a single line.

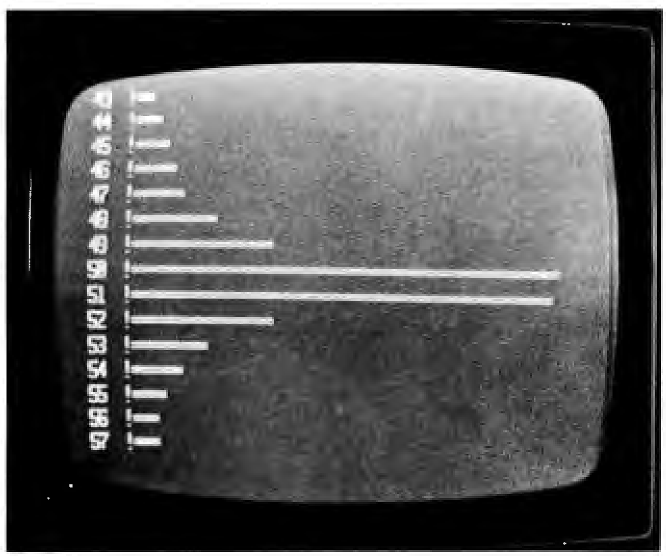


Photo 5: Video graphics display of the magnitude spectrum corresponding to a sine wave whose assumed frequency is 1010 Hz with a sampling interval T = 0.1953 ms. This frequency corresponds to the midpoint between the 50th and 51st harmonics, and the imperfections of the DFT in representing a continuous time signal now can be seen.

Designing a Universal Turing Machine

A Software Approach

Thomas Munnecke
Metasystems
6199 Shaker Dr
Riverside CA 92506

Hardware or software; which is best? This question faces many designers when creating new systems. This article describes a software version of a hardware project detailed in December 1976 BYTE by Jonathan K Millen in his article "A Universal Turing Machine," page 114.

The universal Turing machine (UTM) is elegantly simple and capable of emulating the instruction set of any computer. The Turing machine was invented by Alan Turing (1912-1954). It is an abstract computing device that contains all the fundamental properties a computer system must possess and is used to study computer concepts. Although difficult to program, its back-to-basics nature is alluring to anyone interested in the fundamentals of computers.

The universal Turing machine designed by Jonathan Millen has two memories: one for program storage and the other for the main storage or "tape." The tape is a supposedly infinite (but actually 1024 bits long) memory which is a series of 1s and 0s. A bit on the tape is pointed to by a counter known as the head. A program counter points to a state in the program being executed. Each state consists of two instructions: one to be used if the current bit under the head of the tape is a 1, the other if it is a 0. Each instruction contains fields describing whether to write a 1 or a 0 on the tape, which direction to move the tape (left or right one position), and the address of the next state to be executed.

Each instruction contains the following information:

Bit 0:	Write bit. Write this bit on the tape after the head is adjusted.
Bit 1:	Direction bit. If this is a 0, move the tape to the left; if it is a 1, move the tape to the right.
Bits 2 thru 7:	Next state. These six bits are the number of the next state to be executed.

The reader is referred to Millen's article for a complete description of the universal Turing machine. His design implements this machine with about 15 integrated circuits. The memories are 2102s, the head and program counter are counters, and the control logic consists of various flip flops, shift registers, clocks and decoders. The memories are loaded and examined with switches and a 7 segment light emitting diode (LED). The design is capable of executing about 40,000 instructions per second.

A Software Approach

The program in listing 1 is the logical equivalent of Millen's hardware version for the Motorola 6800 processor. The program storage, tape, program counter and head are parts of the computer's memory set aside for those purposes. The memory organization is shown in table 1. The rest of the logic is programmed via the 6800's instruction set. Table 2 is a comparison of the various functions and their implementation in the two approaches.

The program is a relatively straightforward programming of the hardware version. The basic cycle of functions to be performed is:

- Test the bit on the tape under the head.
- Write a 1 or 0 according to the write bit of the instruction indicated by the program counter and the tape bit.
- Move the tape (adjust the head) according to the instruction's direction bit.
- Set the program counter equal to the address specified by the address bits of the instruction.
- Go back to the first step.

Since the 6800 is a byte oriented machine, the head must keep track of both a byte in memory and a bit within the byte. The

Tinker, Tailor, Soldier, Sailor . . .
Doctor, Lawyer . . . the Chieftain's here.

No matter whether you're a serious hobbyist or a serious businessman, the Chieftain 6800 microcomputer with capabilities that surpass the Z-80 is made for you.

Smoke Signal's quality-packed Chieftain I features two 5.25-inch mini-floppy drives and Chieftain II features two 8-inch floppy drives.

Both microcomputers provide 32K static memory, two serial I/O ports, a 2 MHz processor board, a 2K RAM monitor, a nine-slot motherboard with built-in baud rate generator and gold connectors for high reliability. The Chieftain's stylish leather-grained cabinet houses the above with its own cooling fan and regulated power supply.

Every Chieftain is complete with system software and is totally burned-in as well as tested to further insure high reliability.

And it's expandable to 64K memory with up to 2 megabytes floppy disk storage.

So see your nearest Smoke Signal dealer, he'll be glad to show you how to get your wampum's worth. Systems start at $2,595.

☐ Send information on your Chieftain microcomputer
☐ Send name of nearest dealer

Name ______________________
Address ______________________
Company ______________________
City ______________________
State/Zip ______________________

SMOKE SIGNAL BROADCASTING

31336 Via Colinas, Westlake Village, California 91361, (213) 889-9340

Dealer inquiries invited.

Hail to the Chieftain

Smoke Signal Broadcasting, 31336 Via Colinas, Westlake Village, CA 91361, (213) 889-9340

Circle 320 on inquiry card.

program uses variable HEAD to point to the byte, and MASK to define the bit. MASK consists of seven 0s and a single 1. The 1 corresponds to the bit under consideration. For example, suppose the head now points to the third bit of byte 83 (hexadecimal). HEAD and MASK would then be:

```
HEAD   1000 0011   (binary)
         8    3    (hexadecimal)

MASK   0010 0000   (binary)
         2    0    (hexadecimal)
```

Listing 1: 6800 assembler version of the universal Turing machine, which imitates the hardware version built by Jonathan Millen. This program is capable of executing 10,000 universal Turing machine instructions per second.

```
        Hexadecimal
Address Code       Label    Op Code  Operand   Commentary
                            ORG      $0100
                   PC       EQU      $FB       TURING MACHINE PRO-
                                               GRAM COUNTER
                   MASK     EQU      $FD       SPECIFIES BIT IN HEAD
                                               BYTE
                   HEAD     EQU      $FE       POINTS TO HEAD OF TAPE
 0200   DE FE      BEGIN    LDX      HEAD      GET HEAD OF TAPE ADDR.
 0202   D6 FD               LDA B    MASK      MASKS OUT BIT ON HEAD
 0204   E5 00               BIT B    0,X       IS BIT ON TAPE 0?
 0206   27 03               BEQ      ZEROBIT   YES, DON'T INCREMENT
                                               PROGRAM COUNTER
 0208   7C 00 FC            INC      PC+1      NO, INCREMENT PRO-
                                               GRAM COUNTER
 020B   DE FB      ZEROBIT  LDX      PC        GET ADDRESS OF NEXT
                                               TURING INSTRUCTION
 020D   A6 00               LDA A    0,X       GET TURING INSTRUC-
                                               TION IN REGISTER A
 020F   0C                  CLC                CLEAR CARRY PRIOR TO
                                               TEST
 0210   DE FE               LDX      HEAD      GET HEAD BYTE AGAIN
*SET HEAD BYTE/MASK UP OR DOWN ONE POSITION IF DIRECTION BIT IS 1
 OR 0
 0212   85 40               BIT A    #$40      MASK OFF DIRECTION
                                               BIT
 0214   27 07               BEQ      DEC       DECREMENT IF IT IS
                                               ZERO
 0216   56                  ROR B              ROTATE RIGHT IF IT'S A
                                               ONE
 0217   24 09               BCC      OK        NO CARRY TO NEXT
                                               BYTE
 0219   08                  INX                INCREMENT HEAD BYTE
 021A   56                  ROR B              SHIFT CARRY THROUGH
 021B   20 05               BRA      OK        ALL DONE
 021D   59         DEC      ROL B              ROTATE MASK LEFT ONE
                                               BIT
 021E   24 02               BCC      OK        NO CARRY TO NEXT
                                               BYTE
 0220   09                  DEX                DECREMENT HEAD BYTE
 0221   59                  ROL B              ROTATE THRU CARRY
 0222   DF FE      OK       STX      HEAD      RESTORE HEAD POINTER
 0224   D7 FD               STA B    MASK      RESTORE MASK
*WRITE SPECIFIED BIT ON TAPE
 0226   4D                  TST A              CHECK LEFT BIT OF
                                               INSTRUCTION
 0227   2B 05               BMI      WRITE1    SKIP IF IT IS ON
 0229   53                  COM B              COMPLEMENT MASK TO
                                               WRITE A ZERO
 022A   E4 00               AND B    0,X       'AND' IN A ZERO
 022C   20 02               BRA      BRANCH    SKIP AROUND THE
                                               WRITE-A-ONE LOGIC
 022E   EA 00      WRITE1   ORA B    0,X       'OR' IN A ONE
 0230   E7 00      BRANCH   STA B    0,X       PUT BYTE BACK TO TAPE
*SET TURING MACHINE PROGRAM COUNTER TO NEW ADDRESS
 0232   84 3F               AND A    #$3F      MASK OFF TWO LEFT
                                               BITS
 0234   49                  ROL A              MULTIPLY BY TWO
 0235   97 FC               STA A    PC+1      STORE AS NEW PRO-
                                               GRAM COUNTER
 0237   7E 02 00            JMP      BEGIN     EXECUTE NEXT UTM
                                               INSTRUCTION
```

The logic to test the current bit is:

```
TEST    LDX    HEAD     Load head byte address.
        LDA B  MASK     Load bit mask within byte.
        BIT B  0,X      Test corresponding bit in memory.
        BEQ    ITSONE   Yes, it's a 1.
ITSNOT  ....            logic if bit was a 0.
ITSONE  ....            logic if bit was a 1.
```

The program increments the head position (moves tape to right) by rotating the mask to the right. If the bit is rotated out and into the carry, the HEAD address is incremented. The procedure is similar for moving the tape left.

The universal Turing machine program is stored in the first 128 bytes of memory. Each state consists of two 1 byte instructions, so that the instruction's address in memory is the state number multiplied by 2. The 6800 has no multiply instructions, but in this case the same effect may be accomplished by the rotate left instruction:

```
Before shift  0 0 1 0 1 0 0 1 = decimal 41 state
                                          number.
               / / / / / / /
After shift   0 1 0 1 0 0 1 0 = decimal 82 state
                                          address.
```

Hardware versus Software

Although there is probably not a great practical need for Turing machines of this type, the two designs provide some insights into the benefits and costs of each approach.

The most significant benefit of the hardware approach is speed. The program can only process 10,000 universal Turing machine instructions per second, or 25 percent of the circuit's capability.

The most significant benefit of the software approach is its flexibility. For example, suppose the address field of the instruction

Hexadecimal Addresses	Use
0000 thru 007F	Universal Turing machine program and state storage area.
0080 thru 00FA	Universal Turing machine tape storage area.
00FB thru 00FC	Universal Turing machine program counter address of next state.
00FD	Tape head mask.
00FE thru 00FF	Tape head address.
0200 thru 0237	6800 interpreter program (listing 1).

Table 1: Memory allocation for the software implementation of the universal Turing machine.

More and more, you see the North Star HORIZON computer at work: in business, research, and education. Its high performance qualifies the HORIZON for demanding professional applications. Over 10,000 users during the past two years have proven that North Star hardware has the reliability for day-in, day-out computing. The HORIZON is now a serious candidate for any small system installation.

SOFTWARE IS THE KEY TO HORIZON MATURITY

North Star BASIC and DOS have been used to develop hundreds of commercial program packages. These packages establish that North Star software has the completeness and convenience necessary for serious program development. Because of the many independent vendors offering software using North Star BASIC and DOS, the HORIZON owner now has the widest selection of software in the microcomputer industry! Software available includes: word processing, general ledger, accounts payable/receivable, mailing list processing, inventory and income tax preparation. Program development systems such as assemblers, debuggers, editors, PILOT and FORTRAN are also available.

EXPAND YOUR HORIZON

The basic HORIZON computer includes a Z80 microprocessor, 16K bytes of RAM memory, an I/O interface and one Shugart minifloppy disk drive. The HORIZON can be expanded to 60K bytes or more of RAM, three disk drives, and three I/O interfaces. Performance can be enhanced by the addition of the North Star hardware floating point board. Also, S-100 bus products from other manufacturers may be used to expand the HORIZON.

For more information, contact your local computer store.

NORTH ★ STAR
COMPUTERS

2547 Ninth Street
Berkeley, California 94710
(415) 549-0858

is to represent a signed displacement from the current program counter, as Millen suggests in his article. In hardware, this would require adding a 6 bit adder between the address bus and the program counter, plus some temporary latches to hold the results. In software, a store instruction must be changed to an add instruction. In hardware, the board must be modified to accommodate the new circuitry, and the clock readjusted. In software, under MIKBUG, the change can be made with seven keystrokes.

If this system were to be widely distributed, complete documentation would have to be written. The hardwired approach requires a circuit board layout, a schematic diagram, parts list and written commentary. In the software version, comments in the program serve to document the system, along with a written commentary.

The software approach allows a *building block* technique. The program may be easily combined with other programs. The external programs need to know only the addresses of the various blocks in the universal Turing machine program's logic. The universal Turing machine circuit would have to be modified to adapt it to other equipment. The software version uses MIKBUG's load and dump routines to save the tape contents, but this would have to be a specially constructed circuit for the hardwired design.

The design, implementation and testing times of the software version were two, one and two hours, respectively. I don't know the exact times required for the hardware approach, but they should be at least several times more than the software approach.

In order to build the hardwired circuit, the experimenter must obtain all the circuitry, a circuit board, wire, power supply, etc, which may or may not be used in future experiments. However, once you have a microcomputer to work with, no extra items are needed and the computer is usable for any other projects without losing the ability to reload the universal Turing machine program.

This example cannot be taken as a complete treatment of the trade-offs of the two approaches. Each designer must judge the merits of an approach according to the particular needs of the problem to be solved. If the universal Turing machine ◦ ◦ to be mass-produced for time-critical ◦ ations, the hardware approach would be best. If the design is to be used for the Sunday afternoon project of a microcomputer enthusiast who already has a system the software approach would be best.■

Function	Hardware Version	Software Version
Program storage	2102	Memory locations hexadecimal 00 thru 7F
Tape storage	2102	Memory locations hexadecimal 80 thru FF
Program counter	Two 74161s	Memory locations hexadecimal FB thru FC
Head	Three 74191s, 7474	Memory locations hexadecimal FD thru FF
Sequencing	7404, 74161, 74154	Conditional branching
Display	7 segment LED	MIKBUG print/punch command
Initializing tape	74157, 7400, switch	MIKBUG load command
Saving tape		MIKBUG print/punch command
Debugging design	Logic probe and oscilloscope	MIKBUG break command

Table 2: Correspondence chart of the functions of the two approaches and the means with which they are implemented.

Operating the Turing Machine

1. Put your program in the state storage area. Note that the address of each state is twice the state number.

2. Initialize the tape storage area. You may put your *tape* anywhere in memory as long as you set the tape head pointer to the proper initial address. Location hexadecimal 0080 is convenient.

3. Initialize the program counter. Put 00 in location hexadecimal 00FB, and the first address in the Turing machine in location hexadecimal 00FC. This must be the actual memory address (twice the state number).

4. Initialize the mask. The mask selects which bit of the byte pointed to by the head pointer to operate on. It must be composed of seven 0 bits and a single 1 bit. 01 is a reasonable starting value.

5. Initialize the head pointer. This is the address of the byte in memory to be considered as the head of the tape. It must point to the tape storage area.

6. Set your MIKBUG start address to hexadecimal 0200, press G (for go), and away it goes.

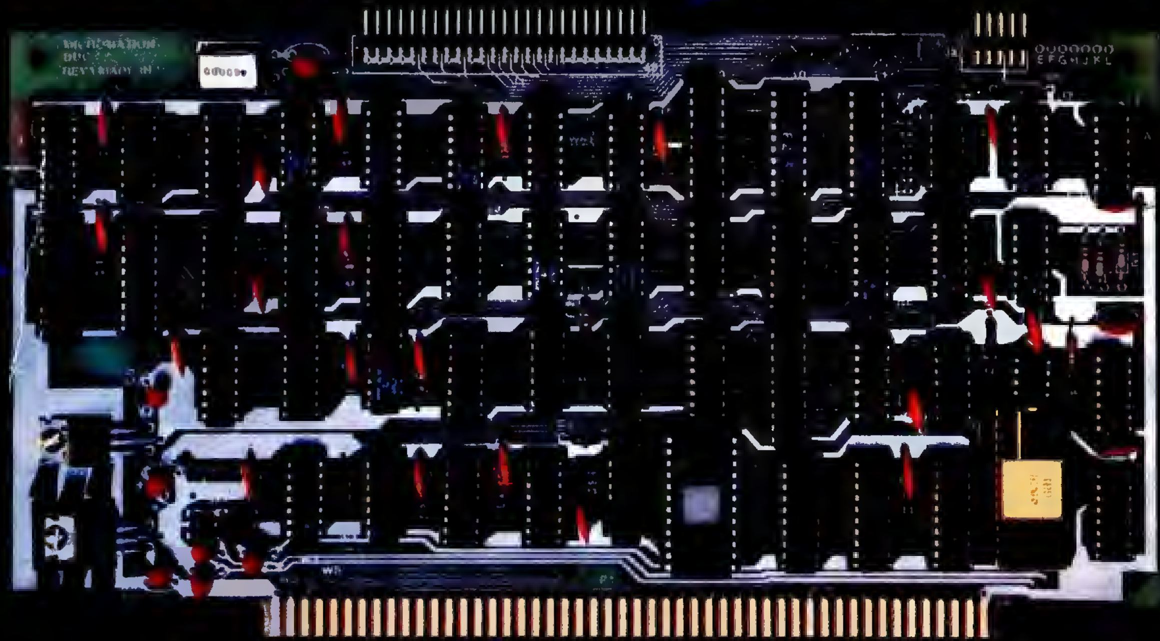
REFERENCES

1. Millen, J, "A Universal Turing Ma
December 1976 BYTE, page 114.

2. Ralston, A, and Meek, C (editors)
pedia of Computer Science, Petroce
New York, 1976, pages 1432 thru 14

3. Shannon, C E, and McCarthy, J
Automata Studies, Princeton Unive
Princeton NJ, 1956.

Micromation has done for the S-100 bus what IBM did for the floppy disk.



Reliably doubled capacity.

Double Capacity

The DOUBLER — Micromation's latest advance in floppy disk technology — doubles the capacity of floppy disk systems. Over 500 KBytes are recorded on each side of an 8" disk. This means bigger files for more powerful systems.

Double Speed

Data transfer with the DOUBLER is twice as fast — 500 Kbits per second. And since there is twice as much data on each track, your drive steps only half as much — so your system runs faster than it ever has before!

Increased Reliability

That's right — even better reliability. Why? Because we did it the IBM way. IBM designed 2D formatting — so it has to be reliable. Micromation's innovative, state-of-the-art design incorporates write precompensation electronics and a phase lock oscillator on a single, all digital, S-100 circuit board. So we guarantee the DOUBLER will be more dependable than your present single density controller — and we warantee the DOUBLER for a full year.

Unbeatable Convenience

It couldn't be easier to step up to double density. The DOUBLER operates automatically in either single or double density. Just insert a diskette and you're running properly. You can transfer files between single or double density diskettes without any software or hardware changes — or even operate with one single and one double density diskette.

Installation is a snap. There's a hardware UART on board and the software is all ready to go. An onboard 2708 EPROM contains the bootstrap. There's even jump-on-reset circuitry so you can operate without a front panel. And, of course, we include utilities to format diskettes.

Universally Versatile

The DOUBLER will operate with all industry-standard mini and full-sized drives. And it will work in any 8080 or Z-80 S-100 computer operating at 2 to 4 MHz. The DOUBLER will support up to four double or single headed drives.

Fully Compatible

The DOUBLER is compatible with CP/M* version 1.4. If you have a CP/M* 1.4 system, just add our CBIOS — or you can buy our ready-to-boot version. Install the new controller, connect any terminal to the RS-232 interface, and boot off your new double-sized, double-speed system. You still can use all your old software without any changes.

Completely Affordable

All Micromation products are fully assembled, thoroughly tested, include complete documentation, and are priced for value:

DOUBLER double density controller	**$ 495.**
MEGABOX dual drive double density system	**2,295.**
ZEPHER — Per Sci double density system	**2,595.**
Z-PLUS — MEGABOX 32 KZ-80 computer	**4,295.**

Available

The DOUBLER is available NOW at your local computer store.

Micromation Inc. 524 Union Street San Francisco California 94133 / 415 398-0289

MICROMATION

Where there's always more in store.

*CP/M is a trademark of Digital Research

Circle 223 on inquiry card.

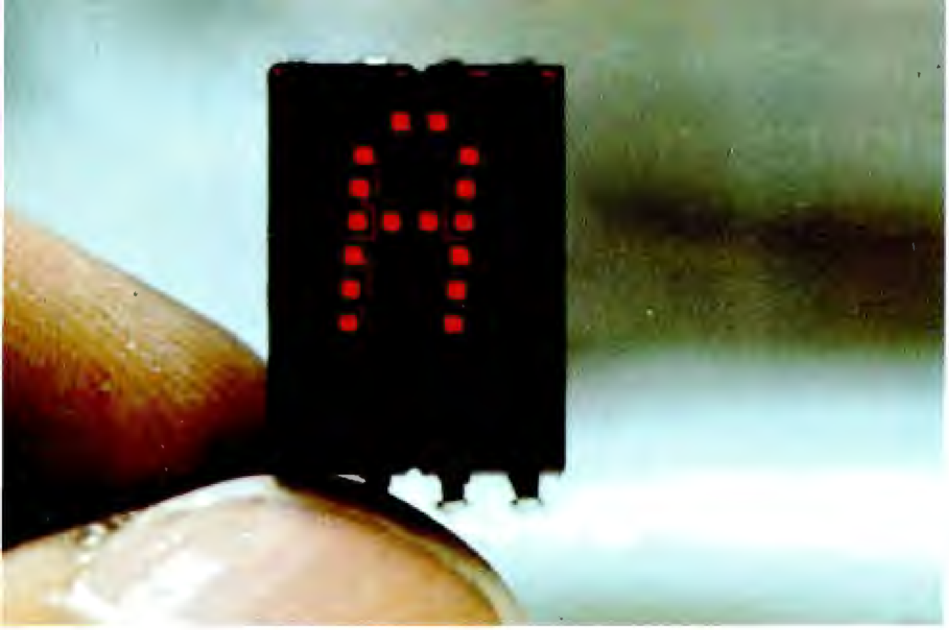


Photo 1: Hewlett-Packard 5082-7340 hexadecimal character display, which uses a pseudo 7 segment dot matrix. On and off control of the dots at the end of each segment allows the circuit to display capital Bs and Ds. The display pictured is powered.

Ciarcia's Circuit Cellar

Build an Octal/Hexadecimal Output Display

Steve Ciarcia
POB 582
Glastonbury CT 06033

"Steve, I think we have a little problem!" Ray charged into the basement and hovered over me waiting for a response.

I slowly rotated in my swivel chair. The rate was barely sufficient to overcome static friction, but I finally made it. As I raised my head to talk I was interrupted.

"Steve, I think we have a problem with that EROM." Before he could finish, his expression abruptly changed and almost without a pause he ended the sentence with, ". . .what happened to you? You look like death warmed over!"

I could barely see the person standing before me with his hands on his hips. I also experienced a strange sensation of either a veil covering my face or an advanced case of furry eyeballs. Whatever the cause, Ray was still standing there awaiting a reply. It was a chore to speak. As the muscles contracted to produce the necessary air flow, I could sense a sudden recurrence of physical problems which I had hoped were on the wane.

"Steve you look terrible! You should be raring to go after two weeks in Acapulco, basking in the sun."

Ray was referring to an engineering consulting job I had just completed in Acapulco for CBS. The Miss Universe Pageant, which was broadcast live from Mexico, included a new twist this year. A computerized judging system. It sounded like a fun consulting job as opposed to the usual, "design me a computer for . . ." type. The final rationalization was, I needed a vacation anyway. I wouldn't want anyone to think that the 70 contestants had anything to do with my decision to go.

The other lucky members of our engineering party were Gus Calabrese (formerly with Digital Group) and George Watson and Dale Walker of CBS. Gus and I maintained the hardware; Dale supported the software; and, while George's official function was the electrical scoring system, his unofficial title was chief taco tester. He had this uncanny ability to sort through all the various smells emanating from a restaurant and evaluate palatability. If he didn't turn green as he walked through the door, it was Amercianized enough for us to eat there.

This smooth sailing trip was punctuated by a succession of daily crises. For instance, George's wife, having thoughtfully packed his suitcase without underwear, gave us the hoped for opportunity to take a crash course in Mexican capitalism and to venture out to the market place. The cab driver who "drove" us there (I use the word loosely) was subject to suicidal fits. From then on everything went downhill. The list goes on and on. Reliving the past two weeks in my thoughts heightened the sense of physical malaise I was experiencing. Fortunately, Ray spoke again in time to bring me back to reality.

"What's wrong with you?"

G/2 MEANS BETTER BASIC, EXCITING COMPUTER GAMES, AND VALUABLE HEALTH HINTS.

Now you don't have to be a programmer to use your computer!

The G/2 Program Library is moving into computer stores all over America. These reliable, professional programs are pre-recorded on cassette tapes. They contain prompting instructions for easy use, and come complete with a detailed instruction manual. G/2's rapidly expanding library of program topics appeals to a broad range of interests, and every program will help make **your** computer a better investment.

G/2 Basics for SOL and SWTPC (created by MICROSOFT™) are far better than the interpreter you're now using. **G/2 Standard Basic** lets your SWTPC 6800 execute programs faster, uses only 7K of memory, and offers string arrays, extensive string functions, peek, poke, wait, continue, calculator execution, 10 nested subroutines, and much more. **G/2 Extended Basic** puts the king of Basic interpreters in your SOL computer. It offers string arrays, 16-digit accuracy, fully descriptive error messages, automatic line numbering and renumbering, an amazing editing capability, trace function, and many additional features you thought were limited to disk-based systems.

G/2 Beat the House brings the excitement of casino-type games of chance right into your own home. When you sit down to play Blackjack, Craps, Roulette or Slot Machine, you'll feel like you're in the middle of a Las Vegas casino, playing for big stakes.

G/2 Clinic helps you understand the effects of environment, heredity, personal habits, diet and even biorhythms on your health. Built-in prompts make these three programs easy for your family and friends to use, even if they've never before touched a computer.

G/2 Contests for Apple Owners use the full power of Apple's color graphics to simulate the tense dangers of interplanetary war, the hair-trigger suspense of a shootout, and the challenges of several mathematical and pure logic puzzlers.

Start building your G/2 Program Library today. For the name of the G/2 dealer nearest you, call us toll-free at 800/538-1770. (In California, call 800/662-9810.) Then load one of these tapes and put the full power of your computer to work for you.

THE REASON YOU BOUGHT YOUR COMPUTER. G2™

GRT CORPORATION

A product of GRT Corporation
Consumer Computer Group
1286 North Lawrence Station Road
Sunnyvale, California 94086
408/734-2910

"Let's just say it has something to do with a guy called Montezuma."

"You're not supposed to drink. . . ."

"Yeah, I know! Don't drink the water!"

Ray looked at me and decided his problem still needed attention, even though I was dying. "Steve, I was about to check the EROM contents against the listing you gave me when I noticed that it was in octal. We need to use that EROM tomorrow and we had better find the error in it tonight. I made a hexadecimal dump of the EROM contents but I still can't check it against your listing."

The response was obvious. "Why don't you convert it by hand?"

"Sure," said Ray, "I can convert it, but a thousand conversions is more than I have time for tonight. Can we assemble it in hexadecimal on your system?"

My temples were starting to throb. I hadn't used my computer in three weeks. Nothing was hooked up and I was in no condition to either attach and fire up my own programmer or write the simple algorithm to perform this minor calculation. It was hard enough for me to remember how to operate my own system without explaining the intricacies to Ray.

"Look, Ray, any night but tonight. I've got it in octal, decimal, hexadecimal, binary, —anything you want, but not tonight. I just don't think I can hack it. You understand, don't you?"

He was disappointed, but being a good friend he understood. "Can I borrow your TI programmer and some desk space? A thousand entries times five button pushes . . . shouldn't take more than an hour or two. Got your battery charger handy?"

It seemed a shame to make Ray go to such lengths. If my system were up it would take only a matter of seconds to print out Ray's listing. It may have been a very powerful Z-80 computer on any other occasion but tonight it wasn't processing anything.

As I reached for the calculator in my briefcase I spied a relic that might provide a solution to the problem. "Ray, see that rectangular box with all the printed circuit boards plugged into the top of it?" I pointed to a bookcase that contained everything but books. "Bring it here and plug it in, and search through that pile of tapes over there until you find one marked with the same name as your listing. I made a binary dump on tape at the same time I made your listing." There are some advantages to being ill—letting others fetch and carry is one of them.

Relying mostly on Ray's high level of hardware expertise, interspersed with whatever limited verbal input I could manage, we successfully fired up my Scelbi-8B 8008 microcomputer. Even though I hadn't used it for well over a year, the read only memory based operating system brought it to life immediately. The recognizable pattern on the light emitting diode (LED) display indicated it was ready to read input data, so I slapped in the cassette that Ray had found. Fortunately the data was stored in a format acceptable by both machines, and totally independent of the processor. I couldn't execute the Z-80 EROM listing I had loaded, but I could display it.

"OK, Ray. Now that we've loaded the data we can step through it on single step and look at it on this output port display, which I built a while back."

"How's that going to help?" Ray looked at the 3 character display as he pressed the single step a few times. "The 8008 is an octal machine. Even the data on your display is coming out octal." he said.

It was hard to smile but I managed a slight variation on the theme as I said, "Flip the switch next to the display." Instantaneously, the 257 previously displayed changed to AF, its hexadecimal equivalent.

"Hey, that's not bad, a combination octal and hexadecimal display! All I have to do is step through and copy down the hexadecimal equivalents, right?"

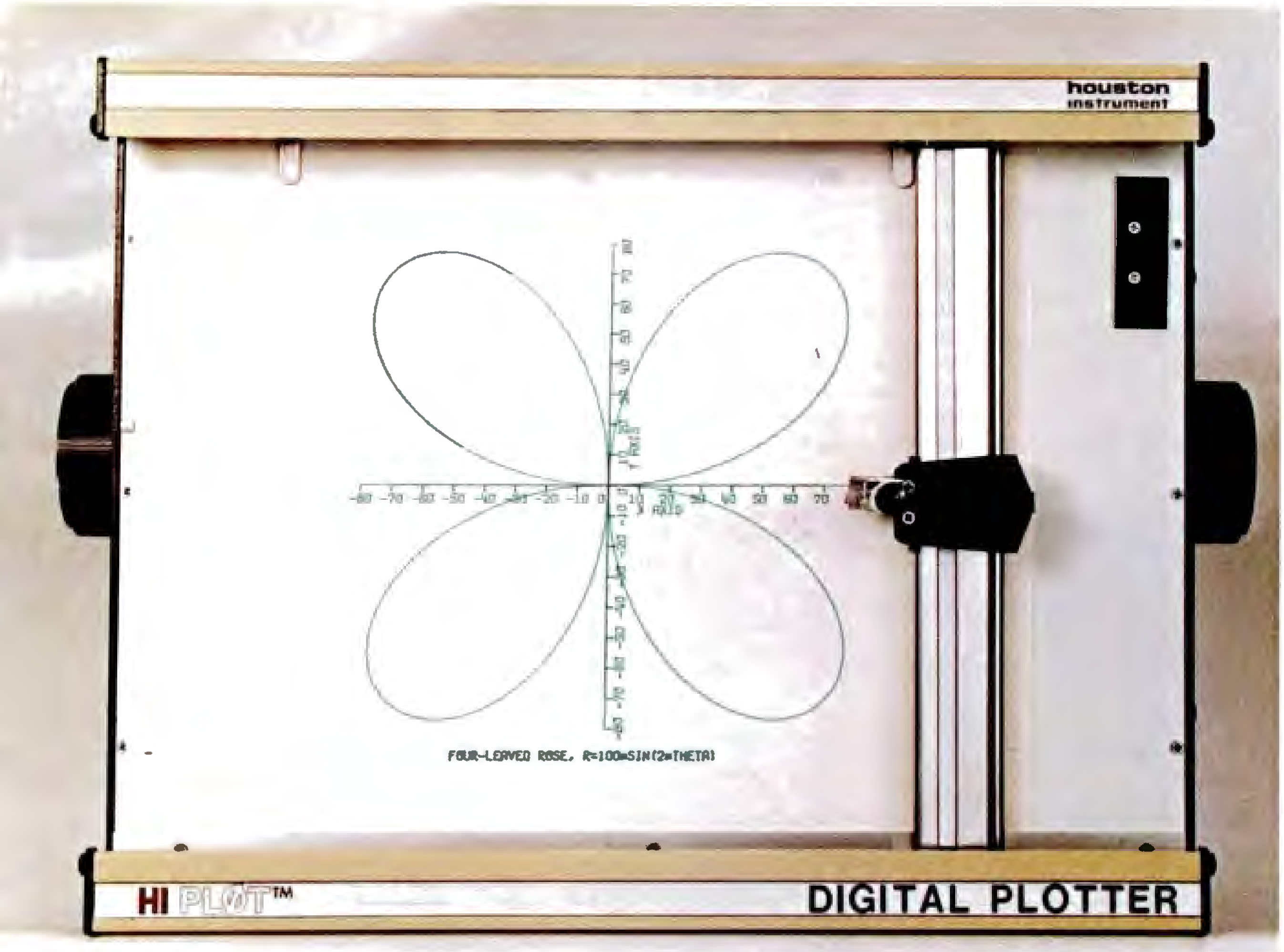
I nodded and Ray started to write. Barely ten entries had been made when his hardware curiosity got the best of him. "I was thinking of putting one of these on my system but it looked like too many components. By the way, I only see two chips. Where are you hiding the rest?"

"Remind me to tell you when I recover."

Build a Combination Octal/Hexadecimal Display

Some people may consider hexadecimal displays a trivial addition to an expensive computer system, but sometimes these little add-ons make program debugging easier. I can't help but wonder whether other computer experimenters would have need for such a display. I don't expect it to replace the video display; but often, when debugging a program, it's nice to be able to display a byte here and there to verify proper program execution. It will never replace the stepper and breakpoint monitor I now use, but it's great to display keyboard or IO data quickly with a single output instruction.

There are many methods to display hexadecimal numbers on a 7 segment LED. Figure 1 and table 1 show an example of the usual brute force method using a read only memory as a hexadecimal decoder. Programming the 82S23 was described in the



The perfect digital plotter for the small system computing enthusiast

Small in size, big in performance...
and priced at only $1085*

HI PLØT ™ is a digital plotter designed for the small systems market. Inexpensive to own and simple to operate, it still incorporates the same quality components, technology and skillful workmanship found in Houston Instrument's larger plotters used throughout industry.

Not a kit, the HI PLØT ™ is assembled and ready to use, with both a RS-232C and a parallel interface build in for creating graphic output from your system.

* U S Domestic Price Only

- Displays data in easy-to-read graphical format
- Both serial and parallel inputs built-in
- Uses standard 8½" x 11" paper
- Plotting speed up to 2.4 ips
- Resolution of both 0.01 and 0.005 inch
- Baud rate and step size easily changed

houston instrument | DIVISION OF **BAUSCH & LOMB**

ONE HOUSTON SQUARE
(512) 837–2820
AUSTIN, TEXAS 78753
TWX 910–874–2022

EUROPEAN HEADQUARTERS Rochesterlaan 6 8240 Gistel Belgium
Phone 059/277445 Telex Bausch 81399

"the recorder company"

For rush literature requests or local sales office information only, persons outside Texas call toll free 1-800-531-5205

ɘ 173 on inquiry card.

November 1975 BYTE ("A Versatile Read Only Memory Programmer," by Peter H Helmers, page 66). While this is a viable approach, an excessive number of components is needed in this stand alone display, and most people would rather not have to program an EROM. The alternatives are to allow a computer to perform the decoding and drive the 7 segment display through the transistors from a latched 8 bit output port, or to put additional logic around a standard 7 segment decoder driver for the extra requirements. The former case necessitates a computer program while the latter can involve as many components as those in figure 1.

Fortunately there are other products on the market that can solve the problem. I've been using the Hewlett-Packard HP7340 hexadecimal display for a number of years. Those familiar with it can rightfully say how trivial the solution was, while those who are not may find it a revelation. Photo 1 illustrates the physical size of the HP7340. A hexadecimal A is displayed in a dot pattern. These hexadecimal displays depart from standard 7 segment format by being capable of displaying a capital B and D in hexadecimal. This is accomplished by controlling the corner dots which give the appearance of "rounding." This ability discriminates a B from an 8 or a D from a 0. There are 16 distinctly different characters.

An additional feature of the HP7340 is that it contains a 4 bit latch and the decoder/driver as well. The result is a single 8 pin hexadecimal display which successfully accomplishes the function of all the circuitry in figure 1. The specification of the individual pins are in figure 4.

Figures 2 and 3 demonstrate how the HP7340 can be configured to function as a 2 digit hexadecimal output port or a 3 digit octal port. No 8 bit latch is required since it

Input Code	82S23 Program	7 Segment Display
D C B A	d7 d6 d5 d4 d3 d2 d1 d0	
0 0 0 0	0 1 1 1 0 1 1 1	0
0 0 0 1	0 1 0 0 0 0 0 1	1
0 0 1 0	0 1 1 0 1 1 1 0	2
0 0 1 1	0 1 1 0 1 0 1 1	3
0 1 0 0	0 1 0 1 1 0 0 1	4
0 1 0 1	0 0 1 1 1 0 1 1	5
0 1 1 0	0 0 1 1 1 1 1 1	6
0 1 1 1	0 1 1 0 0 0 0 1	7
1 0 0 0	0 1 1 1 1 1 1 1	8
1 0 0 1	0 1 1 1 1 0 0 1	9
1 0 1 0	0 1 1 1 1 1 0 1	A
1 0 1 1	0 0 0 1 1 1 1 1	b
1 1 0 0	0 0 1 1 0 1 1 0	C
1 1 0 1	0 1 0 0 1 1 1 1	d
1 1 1 0	0 0 1 1 1 1 1 0	E
1 1 1 1	0 0 1 1 1 1 0 0	F

Table 1: Program for IC2 in figure 1.

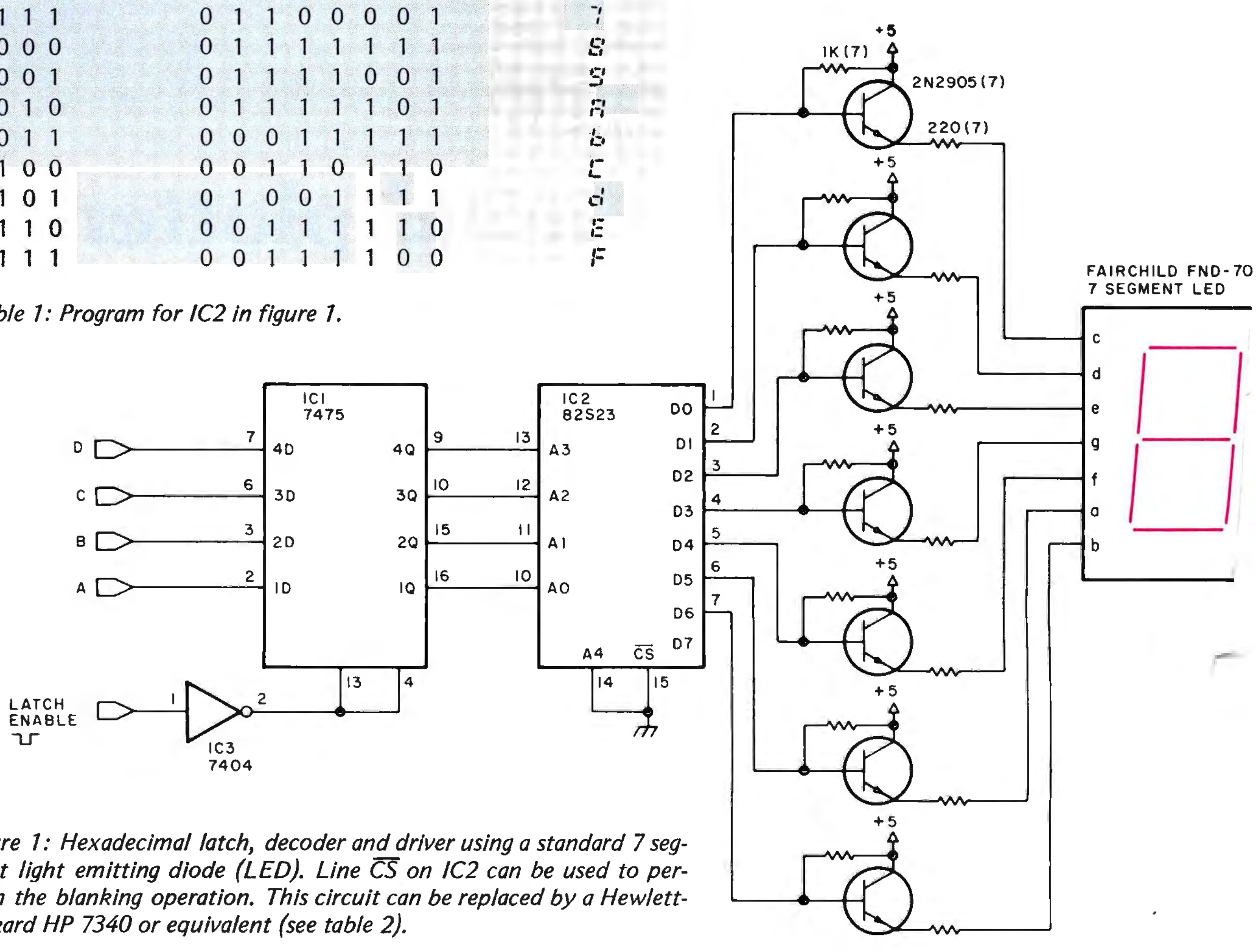


Figure 1: Hexadecimal latch, decoder and driver using a standard 7 segment light emitting diode (LED). Line $\overline{CS}$ on IC2 can be used to perform the blanking operation. This circuit can be replaced by a Hewlett-Packard HP 7340 or equivalent (see table 2).

Circl

Considering a Microcomputer?

Be Sure to Check Out the Product Offerings of the World's Largest Full Line Microcomputer Company.

All Ohio Scientific machines come with microcomputing's fastest full feature BASIC-in-ROM or on-Disk for instant use.

	Minimum Configuration	Base Price
Challenger I Series		
Economical computer systems that talk in BASIC. Ideal for hobbyists, students, education and the home.		
Superboard II — World's first complete system on a board including keyboard, video display, audio cassette, BASIC-in-ROM and up to 8K RAM	4K RAM	$ 279
Challenger IP — Fully packaged Superboard II with power supply	4K RAM	$ 349
Challenger IP Disk — Complete mini-floppy system expandable to 32K RAM	16K RAM	$1190
Challenger IIP Series		
Ultra high performance BUS oriented microcomputers for personal, educational, research and small business use.		
C2-4P — The professional portable	4K RAM	$ 598
C2-8P — The world's most expandable personal machine for business or research applications	4K RAM	$ 799
C2-4P Disk — The ultimate portable	16K RAM	$1464
C2-8P Single Disk — Ideal for education, advanced personal users, etc.	16K RAM	$1738
C2-8P Dual Disk — Most cost effective small business system	32K RAM	$2597
Challenger II Serial Interface Series		
Same great features as Challenger IIP Series for those who have serial terminals: small business, education, industry.		
C2-0—Great starter for users with a terminal	4K RAM	$ 298
C2-1 — Great timeshare user accessory; cuts costs by running simple BASIC programs locally	4K RAM	$ 498
C2-8S — Highly expandable serial machine, can add disks, etc.	4K RAM	$ 545
Challenger III The Ultimate in Small Computers		
The unique three processor system for demanding business, education, research and industrial development applications.		
C3-S1 — World's most popular 8" floppy based microcomputer	32K RAM dual floppys	$3590
C3-OEM — Single package high volume user version of C3-S1	32K RAM dual floppys	$3590
C3-A — Rack mounted multi-user business system directly expandabe to C3-B	48K RAM dual floppys	$5090
C3-B — 74 million byte Winchester disk based system. World's most powerful microcomputer	48K RAM dual floppys	$11,090

OHIO SCIENTIFIC also offers you the broadest line of expansion accessories and the largest selection of affordable software!

Compare the closest Ohio Scientific Model to any other unit you are considering. Compare the performance, real expansion ability, software and price, and you will see why we have become the world's largest full line microcomputer company.

I'm interested in OSI Computers. Send me information on:

- ☐ Personal Computers
- ☐ Small Business Computers
- ☐ Educational Systems
- ☐ Industrial Development Systems
- ☐ I'm enclosing $1.00 for your 64-page small computer buyer's guide.

Ohio residents add 4% tax.

Name ______________________

Address ______________________

City ______________________

State ______________ Zip ________

Phone ______________________

OHIO SCIENTIFIC
1333 S. Chillicothe Road
Aurora, Ohio 44202
(216) 562-3101

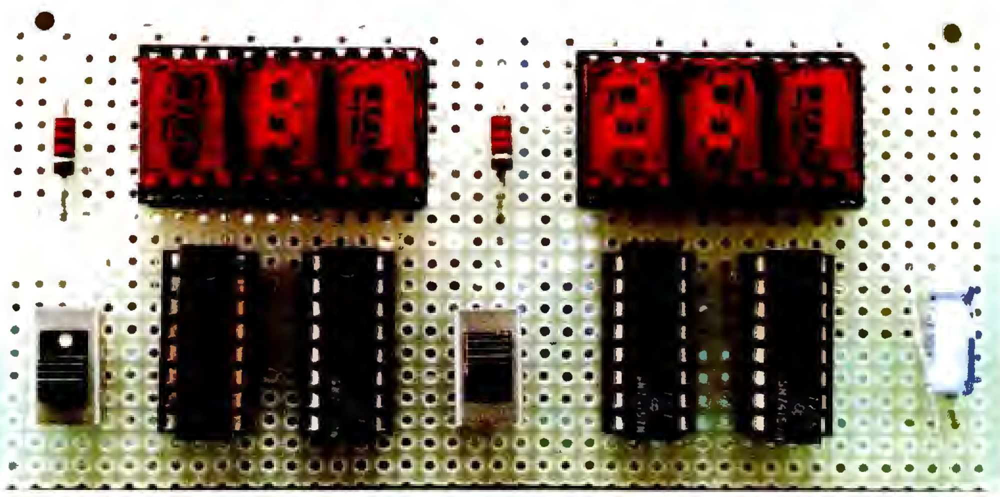


Photo 2: Prototype board of the circuit in figure 4. Two similar circuits were built on the same board. When in the hexadecimal mode (shown at left in the picture), the leading digit is blanked. The display at the right shows the octal mode. Each is wired as an independent output port, but the computer sends the same data to both.

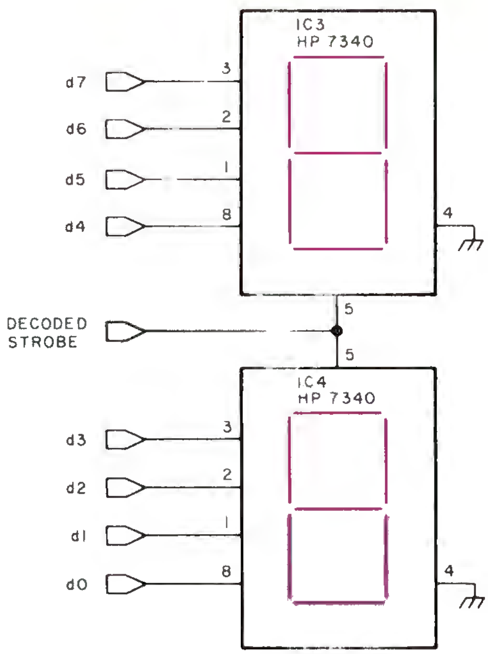
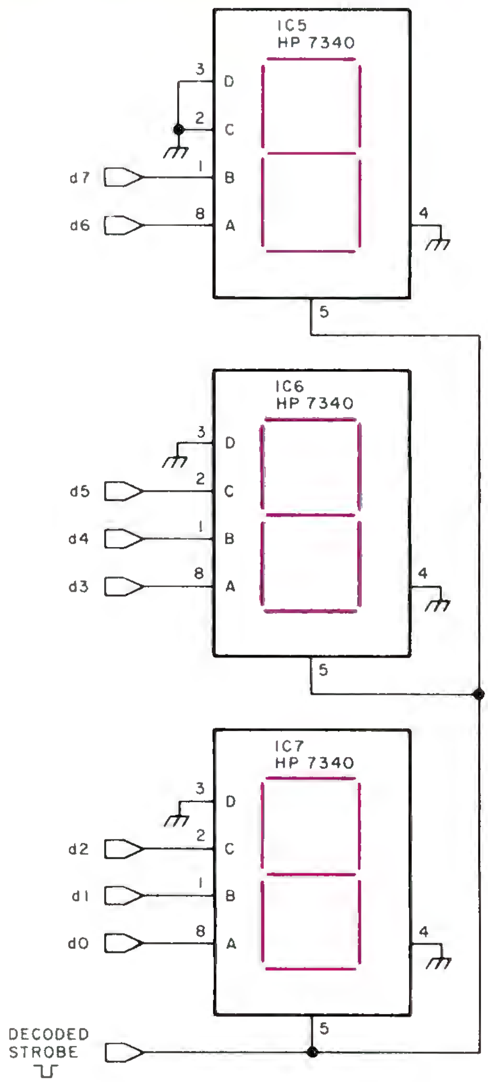


Figure 2: Hexadecimal latch, decoder and driver display circuit.

Pin Number	Function
1	Input B
2	Input C
3	Input D
4	Blank Control (blank = +5 V)
5	Latch enable (latch = 0 V)
6	Ground
7	+5 V
8	Input A

Table 2: Pin functions for the Hewlett-Packard HP7340 binary coded decimal (BCD) to hexadecimal display. Similar displays are made by Dialite and Texas Instruments.



*HP 5082-7300 CAN BE SUBSTITUTED FOR HP 5082-7340 IN OCTAL READOUT APPLICATION. 7300 IS NUMERIC ONLY.

Figure 3: Octal latch, decoder and driver display circuit.

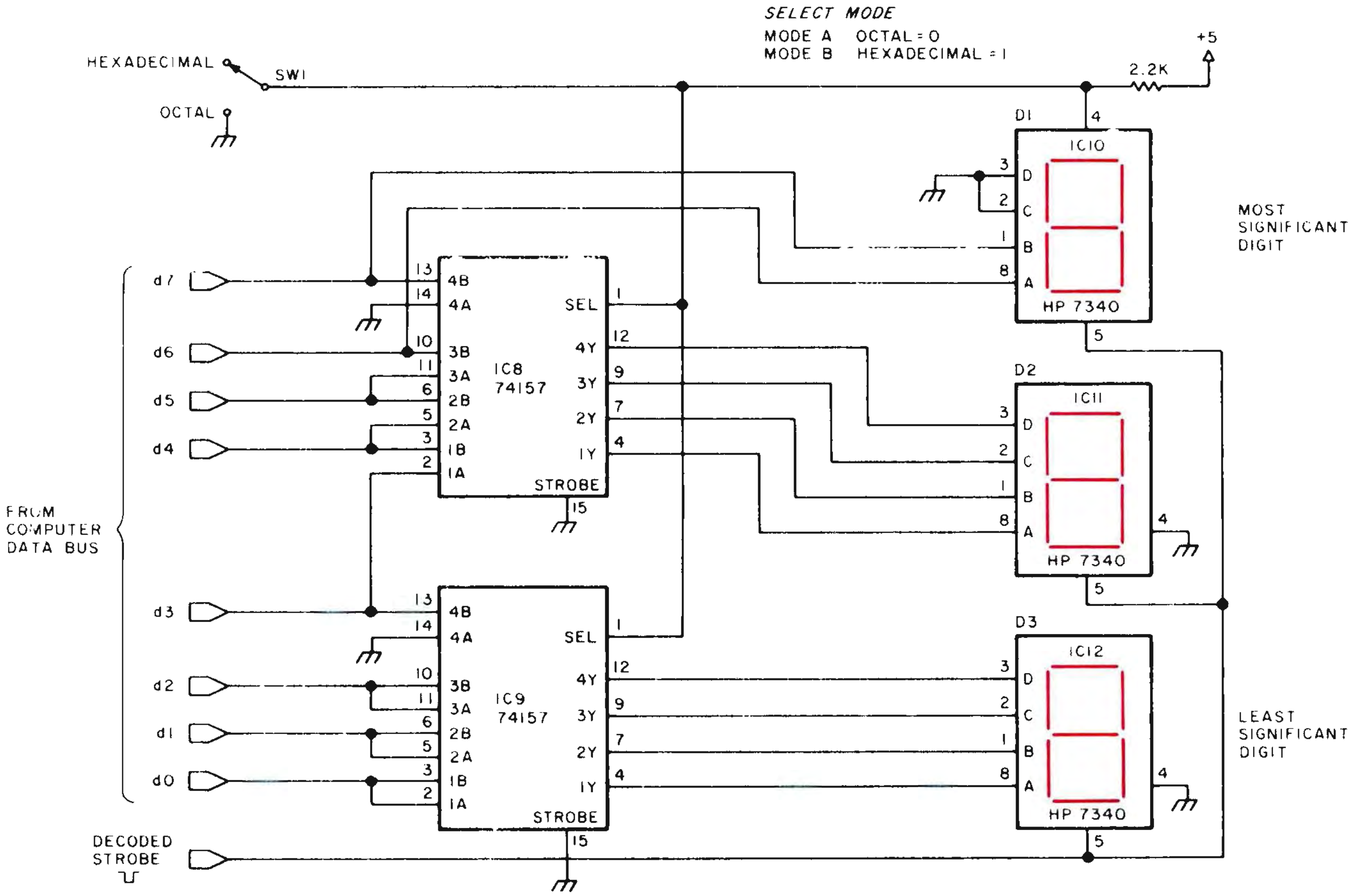


Figure 4: Combination hexadecimal and octal display circuit.

Number	Type	+5 V	Gnd
IC1	7475	5	12
IC2	82S23	16	8
IC3	HP7340	7	6
IC4	HP7340	7	6
IC5	HP7340	7	6
IC6	HP7340	7	6
IC7	HP7340	7	6
IC8	74157	16	8
IC9	74157	16	8
IC10	HP7340	7	6
IC11	HP7340	7	6
IC12	HP7340	7	6

Table 3: Power wiring table for figures 1, 2, 3 and 4.

already contains one. The 7340s can simply be attached to the data bus at any other parallel output port and strobed from a chip select decoder.

Figure 4 is the circuit of the unit similar to the one Ray used. Two multiplexer circuits alternate the input connections to the displays so that when switch 1 (SW1) is in the octal position, the circuit performs as figure 2, and when in the hexadecimal position, as figure 3. The leading character is blanked when in the hexadecimal mode. Two of these circuits are combined in the prototype board of photo 2. The left display is in the hexadecimal mode showing B7 while the right is in the octal mode displaying an equivalent 267 octal. The same binary information is being sent to each port; only the switch setting differs.

Usually these or equivalent displays are advertised only as hexadecimal displays. All strictly hexadecimal displays that I've seen contain these same electronics. While alphanumeric displays will also work, they require extensive scanning logic and are an overkill for this application.

In Conclusion

I hope this simple circuit will eliminate any frustration you may have in the area of hexadecimal displays.

If you have any comments about this or any other article I have written, please write and include a stamped, self-addressed envelope. The mail volume has risen to the point where I'm asked similar questions by many experimenters. A few of these letters will be included each month in BYTE's "Letters" column when appropriate.

One question I'm often asked is whether my introductions are true. So far everything I've written is based upon actual people or events. While I take considerable poetic license in describing the situations, it is not necessary to invent fiction when experience is often so much more humorous.■

Ohio Scientific offers you this remarkable new computer two ways.

Superboard II $279

For electronic buffs. Fully assembled and tested. Requires +5V at 3 Amps and a video monitor or TV with RF converter to be up and running.

Challenger 1P $349

Fully packaged with power supply. Just plug in a video monitor or TV through an RF converter to be up and running.

Standard Features

- Uses the ultra powerful 6502 microprocessor
- 8K Microsoft BASIC-in-ROM
  Full feature BASIC runs faster than currently available personal computers and all 8080-based business computers.
- 4K static RAM on board expandable to 8K
- Full 53-key keyboard with upper/lower case and user programmability
- Kansas City standard audio cassette interface for high reliability
- Full machine code monitor and I/O utilities in ROM
- Direct access video display has 1K of dedicated memory (besides 4K user memory), features upper case, lower case, graphics and gaming characters for an effective screen resolution of up to 256 by 256 points. Normal TV's with overscan display about 24 rows of 24 characters; without overscan up to 30 X 30 characters.

Extras

- Available expander board features 24K static RAM (additional), dual mini-floppy interface, port adapter for printer and modem and an OSI 48 line expansion interface.
- Assembler/editor and extended machine code monitor available.

ORDER FORM

Order direct or from your local Ohio Scientific dealer.

☐ I'm interested. Send me information on your:
  ☐ Personal Computers ☐ Business Systems
☐ Send me a Superboard II $279 enclosed
☐ Send me a Challenger 1P $349 enclosed
☐ Include 4 more K of RAM (8K Total) $69 more enclosed

Name ______________________________

Address ______________________________

City ______________ State ______________ Zip ________

Payment by: BAC (VISA) ____ Master Charge ____ Money Order ____

Credit Card Account # ______________________________

Expires __________ Interbank #(Master Charge) ______________

Ohio Residents add 4% Sales Tax

TOTAL CHARGED OR ENCLOSED ______________

All orders shipped insured UPS unless otherwise requested FOB Aurora, OH

Interested in a bigger system? Ohio Scientific offers 15 other models of microcomputer systems ranging from single board units to 74 million byte hard disk systems.

OHIO SCIENTIFIC

America's Largest Full Line Microcomputer Company

1333 S. Chillicothe Road • Aurora, Ohio 44202 (216) 562-3101

Circle 290 on inquiry card.

WE'RE ALTOS COMPUTER SYSTEMS. Our SUN-SERIES ACS8000 business/scientific computer creates a new standard in quality and reliability in high technology computers.

HIGH TECHNOLOGY The ACS8000 is a single board, Z80®* disk-based computer. It utilizes the ultra-reliable Shugart family of 8 inch, IBM compatible, disk drives. A choice of drives is available: single or double density, single or double sided. Select the disk capacity you need, when you need it: ½M, 1M, 2M, or 4M bytes. The ACS8000 features the ultimate in high technology hardware: a fast 4 MHz Z80 CPU, 64 kilobytes of 16K dynamic RAM, 1 kilobyte of 2708 EPROM, an AMD 9511 floating point processor, a Western Digital floppy disk controller, a Z80 direct memory access, Z80 Parallel and Serial I/O (two serial RS232 ports, 1 parallel port), and a Z80 CTC Programmable Counter/Timer (real time clock). In essence, the best in integrated circuit technology.

BUILT-IN RELIABILITY The ACS8000 is a true single board computer. This makes it inherently reliable and maintainable. The board and the two Shugart drives are easily accessible and can be removed in less than five minutes. All electronics are socketed for quick replacement. Altos provides complete diagnostic utility software for drives and memory.

QUALITY SOFTWARE Unlimited versatility. The ACS 8000 supports the widely accepted CP/M®** disk operating system and FOUR high level languages: BASIC, COBOL, PASCAL and FORTRAN IV. All available NOW.

PRICE $3,840. Standard ACS8000 system with 32 Kb RAM and ½ Mb disk. FPP, DMA and software optional. Dealer/OEM discounts available. Delivery: 3 weeks ARO.

*Z80 is a trademark of Zilog, Inc.
**CP/M is a trademark of Digital Research, Inc.

Dealer/rep inquiries invited.

ALTOS

COMPUTER SYSTEMS 2378B WALSH AVENUE • SANTA CLARA • CA 95050 • (408) 244-5766

Circle 12 on inquiry card.

Book Reviews

The Pocket Calculator Game Book
by Edwin Schlossberg and John Brockman
William Morrow and Company Inc
New York 1975
158 pages hardbound
$6.95

For many of us, the introduction to the microcomputer is the pocket calculator. However, after we learn its functions, it often ends up in a drawer except for shopping and checkbook balancing. It shouldn't. There are many ways to use your calculator for enjoyment.

I like the competitive aspect of this computer business, either against the machine or another player. Therefore I like those books which show me new games to play. In this collection of 50 games, the authors present a variety which will appeal to everyone. There are applications for the "four-banger" as well as the more complex scientific models. You can play with one or more calculators, and one or more players. You may throw in dice or playing cards for variety.

There are easy games and hard ones, offering a range for all ages. Several have two versions, a simple method and complex one for those of you with the costlier machines. "1001" is one of several games whose object is to reach a particular number using the fewest moves. Use dice to determine your move, and hope for luck. There is "Calculator Poker" with betting strategy to guide you. For the business minded, there is "Economy" with all the trappings of high finance. The student of political science will find meat in "Cold War" or "Detente." There are puzzles, mazes and much more. As a puzzle freak, I found my favorite among these pages. By multiplying and dividing in a judicious manner, one can deduce the proper path to follow. A game is a valuable test to see whether your calculator can handle certain operations. This would be an excellent way to check out a calculator that you are considering buying.

Throughout the book, the authors offer samples of how each game should be played, as well as winning strategy. In the introduction are explanations for those concepts which may be new to some readers, such as random numbers, and a glossary of hand calculator terms. A handy index divides the games into like categories, such as number of players, or games with dice, and so forth.

Many games are only the point of departure for exploration by the hobbyist. One could extend many of these to any home built computer. A lot of thought went into this book and it shows. I think the book does the best possible thing for a hobby: it makes the hobby more fun.

Noel K Julkowski
Naval Environmental
Prediction Research Facility
Monterey CA 93940

U.S. POSTAL SERVICE

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION
(Required by 39 U S C 3685)

1. TITLE OF PUBLICATION: BYTE The Small Systems Journal
A. PUBLICATION NO.: 1 0 2 4 1 0
2. DATE OF FILING: Oct. 29, 1978
3. FREQUENCY OF ISSUE: Monthly
A. NO. OF ISSUES PUBLISHED ANNUALLY: 12
B. ANNUAL SUBSCRIPTION PRICE: $15/year
4. LOCATION OF KNOWN OFFICE OF PUBLICATION *(Street, City, County, State and ZIP Code) (Not printers)*: 70 Main Street, Peterborough, NH 03458
5. LOCATION OF THE HEADQUARTERS OR GENERAL BUSINESS OFFICES OF THE PUBLISHERS *(Not printers)*: 70 Main Street, Peterborough, NH 03458
6. NAMES AND COMPLETE ADDRESSES OF PUBLISHER EDITOR AND MANAGING EDITOR
PUBLISHER *(Name and Address)*: Virginia Londoner, Grove Street, Peterborough, NH 03458
EDITOR *(Name and Address)*: Carl T Helmers, Jr., Noone Avenue, Peterborough, NH 03458
MANAGING EDITOR *(Name and Address)*:

7. OWNER *(If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be given.)*

NAME	ADDRESS
Virginia Londoner	Grove Street, Peterborough, NH
Carl T Helmers, Jr.	Noone Avenue, Peterborough, NH

8. KNOWN BONDHOLDERS, MORTGAGEES, AND OTHER SECURITY HOLDERS OWNING OR HOLDING 1 PERCENT OR MORE OF TOTAL AMOUNT OF BONDS, MORTGAGES OR OTHER SECURITIES *(If there are none, so state)*

NAME	ADDRESS
None	

9. FOR COMPLETION BY NONPROFIT ORGANIZATIONS AUTHORIZED TO MAIL AT SPECIAL RATES *(Section 132.122, PSM)*
The purpose, function, and nonprofit status of this organization and the exempt status for Federal income tax purposes *(Check one)*

☐ HAVE NOT CHANGED DURING PRECEDING 12 MONTHS ☐ HAVE CHANGED DURING PRECEDING 12 MONTHS *(If changed, publisher must submit explanation of change with this statement.)*

10. EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES PRINTED *(Net Press Run)*	133,765	139,400
B. PAID CIRCULATION 1. SALES THROUGH DEALERS AND CARRIERS, STREET VENDORS AND COUNTER SALES	21,469	22,128
2. MAIL SUBSCRIPTIONS	103,484	110,464
C. TOTAL PAID CIRCULATION *(Sum of 10B1 and 10B2)*	124,953	132,595
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	3,423	3,817
E. TOTAL DISTRIBUTION *(Sum of C and D)*	128,376	136,412
F. COPIES NOT DISTRIBUTED 1. OFFICE USE, LEFT OVER, UNACCOUNTED, SPOILED AFTER PRINTING	3,710	2,988
2. RETURNS FROM NEWS AGENTS	1,679	none as of today
G. TOTAL *(Sum of E, F1 and 2—should equal net press run shown in A)*	133,765	139,400

11. **I certify that the statements made by me above are correct and complete.**
SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER: Publisher

12. FOR COMPLETION BY PUBLISHERS MAILING AT THE REGULAR RATES *(Section 132.121, Postal Service Manual)*

39 U. S. C. 3626 provides in pertinent part: "No person who would have been entitled to mail matter under former section 4359 of this title shall mail such matter at the rates provided under this subsection unless he files annually with the Postal Service a written request for permission to mail matter at such rates."

In accordance with the provisions of this statute, I hereby request permission to mail the publication named in Item 1 at the phased postage rates presently authorized by 39 U. S. C. 3626.

SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER: Publisher

PS Form 3526 Mar. 1977 *(Page 1)* (See instructions on reverse)

COSMAC VIP

$249* gets the entire family into creating video games, graphics and control functions. For starters.

COSMAC VIP, the completely assembled, ready-to-operate RCA Video Interface Processor, opens up a whole new world of computer excitement. New challenges in graphics, games and control functions. Yet it's just $249.00.

Easy to buy. And easy to program, thanks to its unique, easy-to-use interpretive language. You get a complete how-to book including programs for 20 games: fun, challenging, and ready to load and record on your cassette.

Simple but powerful.

Built around an RCA COSMAC microprocessor, the VIP is a complete computer system that can grow with you. It has 2K of RAM, expandable on-board to 4K. Plus a ROM monitor, audio tone output to a built-in speaker, power supply, and 8-bit input and output ports for control of relays, sensors, or other peripherals.

Soon RCA will offer options for color graphics and 256 tone sound generation. An optional auxiliary keyboard will open up an exciting world of two-player games.

Take the first step now.

Check your local computer store or electronics distributor for the VIP. Or contact RCA VIP Marketing, New Holland Avenue, Lancaster, PA 17604. Phone (717) 291-5848.

*Suggested retail price. Does not include video monitor or cassette recorder.

The fun way into computers.

RCA

Circle 322 on inquiry card.

Life With Your Computer

Justin Milliun
Judy Reardon
Peter Smart
W Ossipee Rd
Silver Lake NH 03875

What can you do with your computer? After hearing about the game of Life, you may never ask the question again. Within the capabilities of a very minimal system, Life gives the computer the kind of job it does best: an enormous amount of repetitive logical operations. *[The authors' system demonstrates this point: it had 2 K memory and a video terminal at the time of this writing.]* This leaves you, the user, free to apply your creative energy on this fascinating game.

Developed by John Horton Conway, a British mathematician at the University of Cambridge, Life was first described in the October 1970 *Scientific American* by Martin Gardner in his "Mathematical Games" column. Its name comes from its resemblance to changing societies of living organisms which can grow, move and occasionally die out.

The Game of Life

An easy way to understand this game is to imagine an immense gridwork or checkerboard. We call each square in the checkerboard a cell, and the entire board a cellular space. Each cell is identical and can perform a number of specific functions. We won't worry about the edge of the board: let's say the space is large enough so that we never know there is an edge. In the game of Life, each cell can sense its eight neighboring cells (as in figure 1). Each cell in our space is in one of two states: it is either alive, or not alive (quiescent). The cellular space changes with time; time advances over the entire space at once, in steps. Each of these steps is called a generation.

The rules which determine the state of a given cell in the next generation are what give Life its delightful properties. They were chosen with great care by Conway, with reasons in mind that will be discussed later. *[For mathematical background information see the book,* Introduction to Artificial Intelligence *by Philip C Jackson, published in 1974 by Petrocelli-Charter. A discussion of cellular automata and pointers to several detailed references are found in chapter 8.]* Let us say there is a pattern of cells in the cellular space, some living, some not. The rules tell which presently living cells survive, which living cells die, and which cells that are not now alive will be living in the next generation. The rules are as follows:

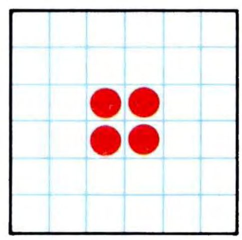
- Each cell presently alive which has either two or three of its eight neighboring cells alive will be living in the next generation.
- Each cell presently alive which has other than exactly two or three live neighbors will not be alive in the next generation.
- If a cell is presently not alive, and exactly three of its eight neighboring cells are alive, it will be living in the next generation.

The above rules are applied all at once in the program for the game of Life. Every cell in the space is checked, as are its neighbors. The fate of that cell in the next generation is then determined. Note that this will amount to many thousands of checks in each generation for a cellular space filling even a small video display screen.

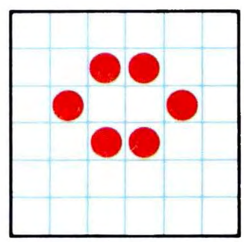
When the program has been loaded into the computer and you've entered the pattern, what can happen as the pattern evolves? There are a number of possibilities.

1	2	3
4	0	5
6	7	8

Figure 1: The center cell (0) has eight neighbors, as does every cell except those bordering the edges of the cellular space in any finite buffer in a computer program. Treating boundary conditions for a finite Life buffer is a fine point of Life program design.



2a



2b

Figure 2: Examples of still life cell patterns, the block (a) and the beehive (b).

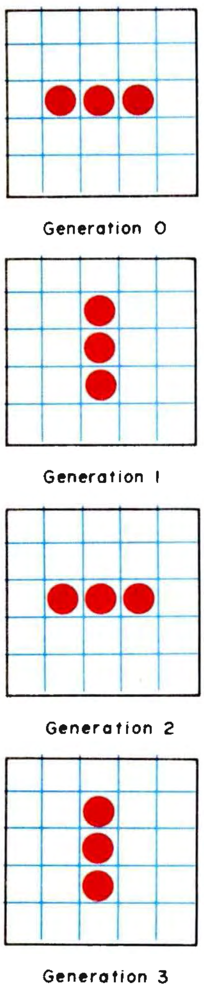


Figure 3: Four generations of the blinker pattern.

Types of Patterns

The pattern may die, leaving you with an empty display as you search your imagination for another possibility to try.

It may stop at what Conway calls a still life. A simple example (figure 2) is the block, another the beehive. These patterns, when left undisturbed, remain the same generation to generation (a little more interesting than a blank screen, perhaps).

The pattern may develop a repeating cycle. The simplest of these is the blinker, which returns to its original self every other generation (see figure 3). A more sophisticated periodically repeating pattern has been described in the February 1971 *Scientific American*. Discovered by G D Collins Jr, it is called the tumbler (figure 4). It has a period of 14 generations, but after seven it is an upside-down copy of the original pattern; hence its name. Watching the tumbler change, you will notice that in every generation there is a row of empty cells separating two mirror image patterns. Each half helps keep the other half under control. If left to itself, half a tumbler will run over 100 generations before settling down.

There are patterns which have most intriguing properties of motion. The glider shown in figure 5 is one such pattern. It is so named because the way that it moves is called glide reflection, or reflection from a diagonal line. In four generations the glider produces a replica of itself, facing the same direction but displaced one square diagonally. After only two generations, it is a copy of itself pointing 90° from its original orientation. There are actually just two unique patterns in the life history of the glider, but it takes four generations for the orientation to match that of the original pattern.

Another example of a moving pattern is the lightweight spaceship shown in figure 6. This pattern also requires four generations to move and to complete a full cycle; it also has only two unique patterns if we disregard their orientation. Note that this pattern moves along a line of cells, as opposed to the glider's diagonal motion.

Finally, patterns exist which continue indefinitely, forever evolving. (It was not certain that such infinite patterns should exist for some time after Life was developed.)

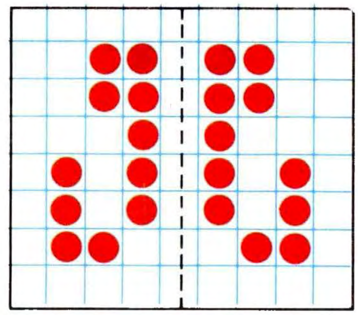


Figure 4: An example of the tumbler, a periodically repeating pattern.

Infinite Evolution a Possibility?

Conway selected the rules of Life to meet the following considerations:

- It should not be obvious that an initial pattern will grow without limit. (Conway specified that cells can die from overcrowding.)
- It should seem possible (but not obvious) that some patterns will grow without limit.
- Some initial patterns should grow and change for a considerable period of time.

As reported in the original *Scientific American* series of articles, Conway conjectured that there were no patterns which would actually grow without limit. At that time he offered a $50 prize to the first person to prove or disprove his conjecture. A short time afterwards a group from MIT disproved it by their discovery of a glider gun. The glider gun produces a glider every 30 generations. Since the glider moves away from its birthplace, we may consider the glider gun to be a special type of repeating pattern (see figure 7). The MIT group also discovered other remarkable events in Life by observing such things as collisions between numerous gliders.

Symmetry

A rather curious property in the evolution of many patterns is their tendency to gain symmetry. As an example, let us begin with a pattern with only partial symmetry called the snowflake, shown in figure 8a. In 15 generations it becomes the pattern called

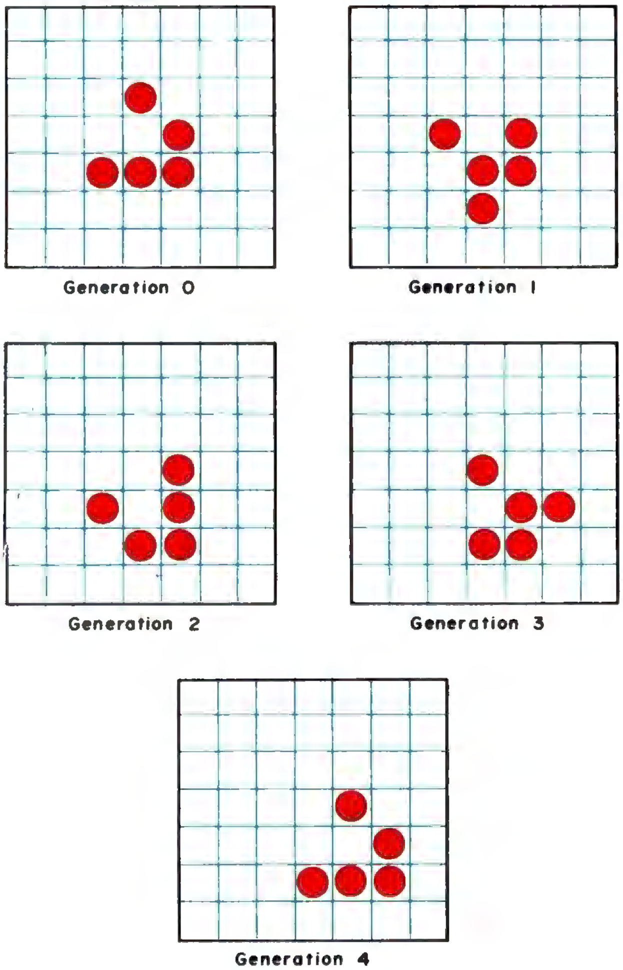


Figure 5: Four generations of a glider pattern. Note that in generation 2, the glider is reflected and rotated 90°, moving down one space. In generation 4, the glider is again reflected, but rotated 90° in the opposite direction, as well as moved over one space.

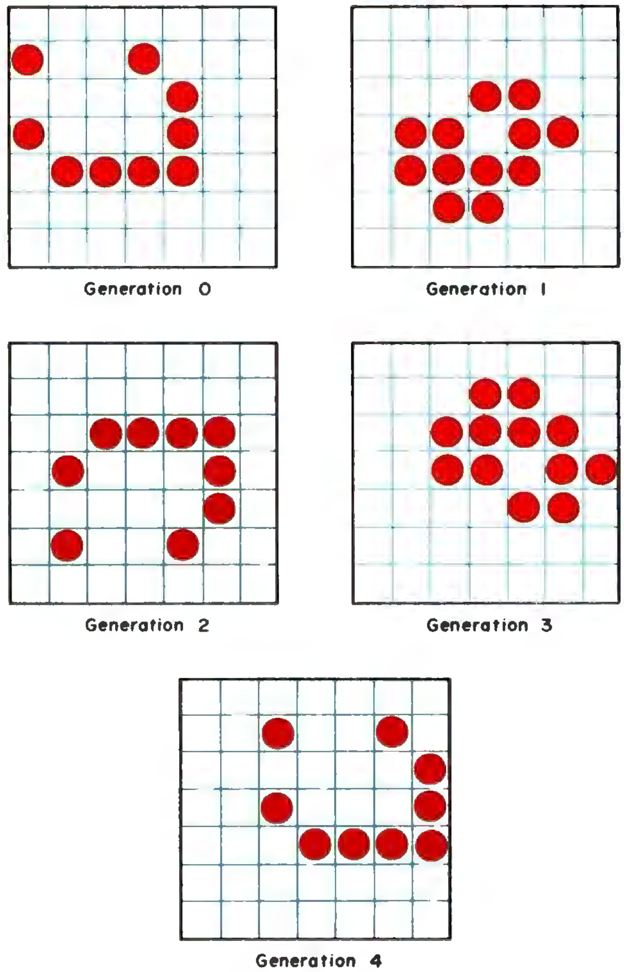


Figure 6: This example of a lightweight spaceship pattern also has a 4 generation pattern cycle. Note that generation 2 is a 90° rotation of generation 0 and is moved over one space. In generation 4, the spaceship returns to the orientation of generation 0, but has moved over two spaces.

the honey farm, shown in figure 8b. (This initial pattern used for producing a honey farm differs from the one discovered by Conway and his collaborators.) Within the brief history of this pattern, it gains new symmetry it can never lose. Apparently, unless the pattern dies out completely, symmetry can only be gained, never lost.

Another example of a pattern increasing its symmetry is given in figure 9. Again, beginning with only partial symmetry, it evolves into another known pattern, a beautiful oscillating pattern called Pulsar CP 48-56-72. It reaches this pattern in 26 generations during which it gains its symmetry. (The initial pattern given here is again an alternative path to the known pattern.) The oscillating period of the Pulsar is three generations, and it provides a very interesting display.

The "Garden of Eden"

Up to now, the evolution of patterns has been considered, using some initial pattern. Can this initial pattern itself have unusual properties? It has been proven that a so-called "Garden of Eden" pattern must exist for the game of Life. A "Garden of Eden" pattern is one which cannot be produced by any other pattern. In other words, no pattern ever becomes a "Garden

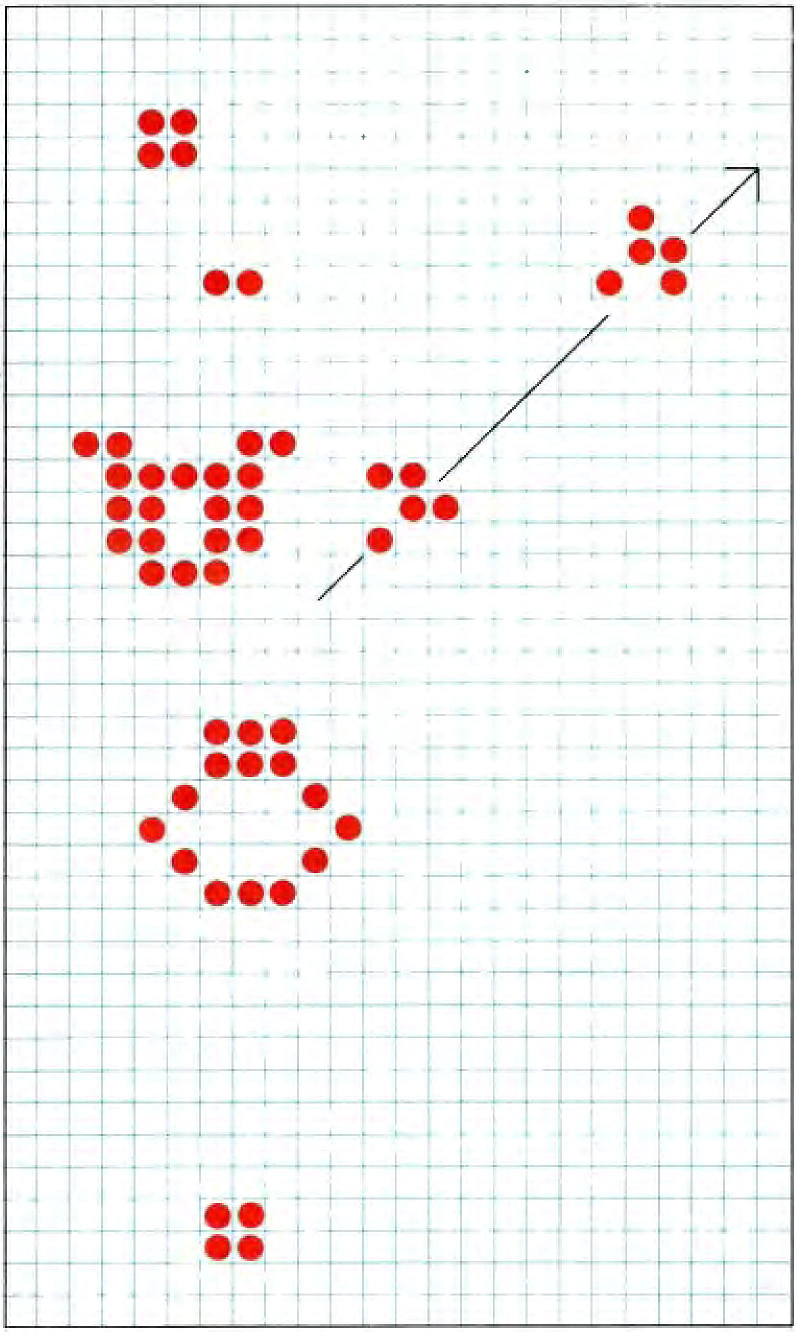


Figure 7: The glider gun is shown in mid-cycle with two generated gliders, the arrow indicating their direction of travel. A more detailed description of this pattern is given in the February 1971 Scientific American *column by Martin Gardner, page 114.*

of Eden" pattern. Such a pattern can only exist, therefore, as an initial pattern specified in generation zero.

It has been shown that a "Garden of Eden" pattern for Life can exist within a square with 10 billion squares on each side, but possibly it could be much smaller.

Perhaps one way to find a "Garden of Eden" pattern is to apply some programming skill, and all your computer's spare time. You start with a pattern you devised or one your computer generated, randomly or otherwise. The computer could then attempt to find a pattern which generates your pattern by trying out all the patterns possible in a larger space than your pattern uses. If it did not succeed in generating your pattern, you would have found a "Garden of Eden" pattern. A lucky application of this brute force technique may come up with the answer, but the going will be very slow. Perhaps some clever shortcut could be developed.

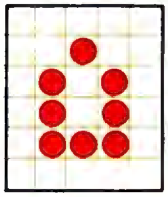
A Warning

The game of Life is downright addictive because there is always another pattern that you'll want to try out in your search for attractive, unusual or unpredictable patterns. Examining properties of symmetry and motion, and looking for "Garden of Eden" patterns and patterns like the glider gun will test your ability to predict fate in the game of Life.

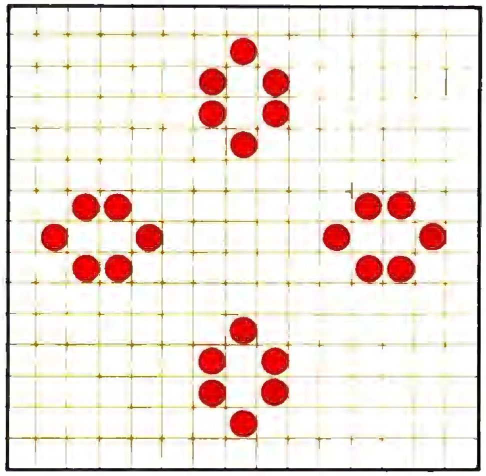
The Program

A program for Life could be written in BASIC or some other high level language, but it would be grossly inefficient, both in size and speed. First of all, only a single bit is required to store each cell, but there is no simple way to manipulate individual bits in BASIC. You would therefore be forced to use one element of a floating point array for *each* cell, and since a floating point number typically uses three bytes, you would only make use of one out of every 24 bits. That would mean for a 64 by 64 cellular array you would need 12 K bytes. (Actually, you would need twice this number since two copies of the cellular array are needed in the simple Life program to be described.) The obvious way to reduce the size is to use every bit; a 64 by 64 cellular array would then require only 512

Figure 8: The snowflake (a) and the pattern it generates after 15 generations, the honey farm (b).

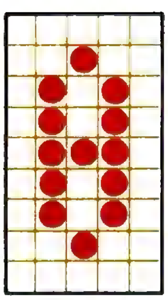


8a

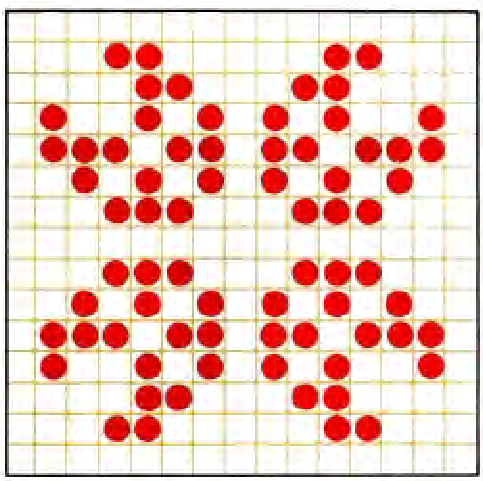


8b

Figure 9: This seed pattern (a) generates pulsar CP 48-56-72. One of the four cycle phases of the pulsar is shown in (b).



9a



9b

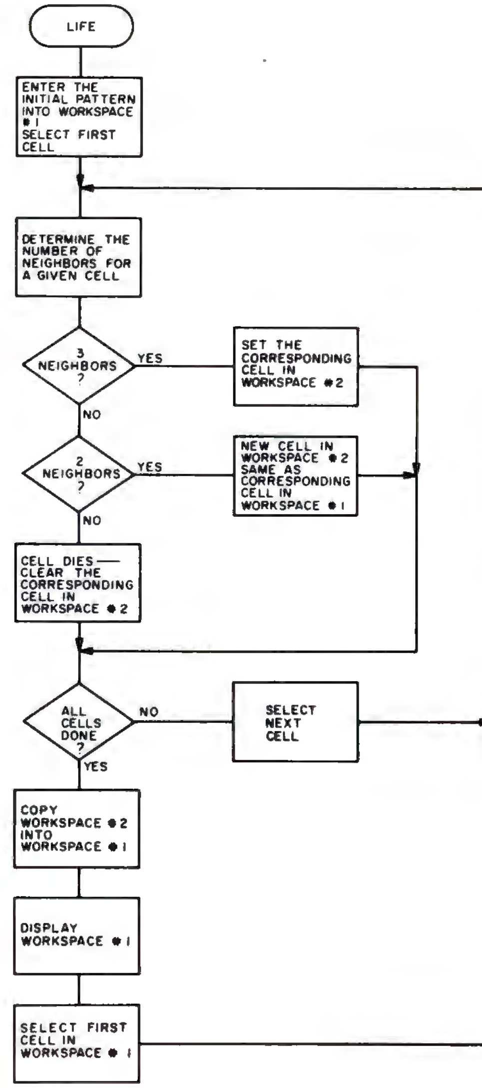


Figure 10: Flowchart of a Life program. [**Authors' note:** *We have written a Life program for the SwTPC 6800 system from this flowchart. Our version requires about 1.5 K of memory for a 2000 cell array, and takes only 6 seconds per generation. Our program can be configured for any size array up to a maximum of 2016 cells. Our 6800 Life program can be obtained from The Computer Warehouse Store, 584 Commonwealth Av, Boston MA 02215.*]

bytes. The most efficient way to manipulate individual bits is to use assembly language. This will also be about ten times faster than a corresponding BASIC program, due to direct execution.

A general flowchart for a Life program is shown in figure 10. The program requires two workspaces, each the full size of the desired cellular array. The initial pattern is entered into workspace #1. The program then creates the next generation in workspace #2, since the original generation must not be altered until the new one is completed. After determining the new generation in workspace #2, it is copied back into workspace #1. The pattern is then displayed, and the procedure is repeated until the program is stopped.

As can easily be seen, Life can be a very interesting game. There are certainly moving patterns other than those described herein, and the possibility exists that one of you reading this could find a previously unknown moving pattern. If you make any discoveries while running Life, we would be glad to hear from you. Enjoy Life!■

MICRO WORLD DECEMBER SPECIALS

Double density Horizon Kit

just announced . . . double the storage at the same price!

The best buy just got better. Now, Horizons are double density at the same price as single density. 180K bytes of on-line information per disk! Single density still runs on your new Horizon, or you can copy and convert all North Star software and programs to double density.

- 180K bytes per disk
- Z-80 processor
- 16K RAM
- add'l 16K memory, [illegible]
- add'l disk drive, [illegible]
- exclusive application software

$1349
(reg. $1599)

NEW!

Call for low assembled prices. Double density also available on North Star disk sub-system, [illegible] kit.

LA 180 DECprinter with Serial Interface

immediate delivery on the field-proven performer

180 cps has never been so affordable. For a limited time, MicroWorld lowers the price of Digital's versatile medium speed printer. Reliable technology and an extensive array of human engineering features make the LA 180 the smart choice for local or remote business applications.

- 180 characters per second printing
- tractor feed with switchable forms length control
- upper/lower case & stand
- compressed print, 16.5 characters per inch

$1895
(reg. [illegible])

Texas Instruments Impact Printer

immediate delivery

TI's new 810 multi-copy impact printer. Prints up to 440 lines per minute because of its unique look-ahead bi-directional feature, controlled by an on-board microprocessor.

- 150 cps, 110 to 9600 baud switchable
- includes tractor feed
- RS232 serial interface

$1695

IP-125 Printer by Integral Data

$749
(reg. [illegible])

More specials from stock

Soroc IQ 120	$ 859
TI 745 KSR Portable Terminal	$1650
Hazeltine 1400 Terminal (ass'l'd)	$ 770
Hazeltine 1500 Terminal (ass'l'd)	$ 995
Bally Arcade	$ 299
Centronics 779 with tractors	$1095

MICRO WORLD™
1425 W. 12th Pl. • Tempe, AZ. 85281 • 602-894-1193
or place your order TOLL FREE
1-800-528-1418

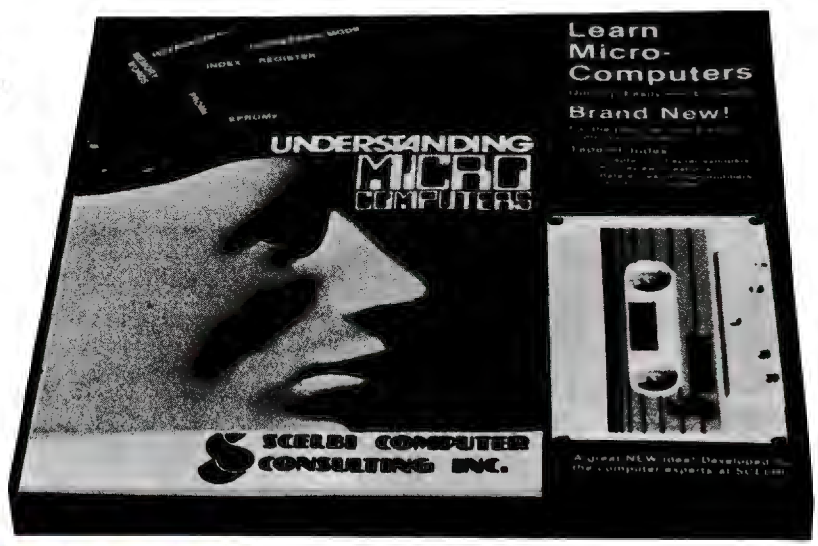
NEW! MicroShopper Guide Vol. V $5.00 (includes $1.05 postage)
MicroWorld (formerly Byte Shop Mail Order) is a division of The Phoenix Group, Inc. Call for all your computer needs.
all prices plus shpg. & hdlg.

Circle 226 on inquiry card.

ALWAYS THE SAME PRICE FOR VISA & M/C — CALL TODAY!

SCELBI MAKES

You can rely on SCELBI publications. Becaus
Computer info that is fun and exciting.
Written and illustrated with YOU, the compu



NEW! ***"Learn Micro-Computers."* Quickly, easily and enjoyably. For the beginner just getting started in microcomputers.**

Here are all the fundamentals behind the operation of virtually all micro-computers. Everything the beginner needs in one handy package. Contains the popular book **Understanding Microcomputers.** Written in simple English. Clear. Concise. Accepted as the standard for the neophyte, you must own this 300-page no-nonsense, easy-reading text. Includes simple-to-use glossary of key microcomputer words. Companion cassette helps you get right into it. Chapter-by-chapter talking synopsis. Key review questions. References to page numbers for further review. You'll gain extra knowledge needed for reading and understanding computer magazines, manufacturer's literature and to feel "at home" around computers. A great new idea developed by the computer experts at SCELBI. Get Yours Now! At your favorite computer electronics dealer. Or order direct. **Learn Micro-Compute** only **$14.95.** Paperback edition of book alone: **$9.95.** Hardcover edition alone: **$14.95.** Shipping handling extra.* (Available after December 1)

Some Facts of Life

David J Buckingham
Computer Communications
Network Group
E4, room 2369A
University of Waterloo
Waterloo, Ontario
CANADA N2L 3G1

Introduction

Life is a game that was developed by Prof John H Conway at the University of Cambridge and first presented by Martin Gardner in the October 1970 "Mathematical Games" column in *Scientific American*. The game is derived from a field of mathematics known as *automata theory* (in this case *cellular automata*). In the February 1971 "Mathematical Games" column the game was described again along with a good introduction to automata theory.

The game is played on a uniform cellular grid (in this case an area divided into squares, such as graph paper) where every cell is surrounded by eight immediate neighbors (ie: cells touching the center cell under consideration). Each cell, or *automaton*, can be in either a 1 or 0 state (on or off – alive or dead). The population of cells is changed by a set of predetermined rules. These changes proceed in intervals called *generations*.

The rules are as follows:

- If a live cell is surrounded by two or three live cells in the present generation, it will remain on (or live) in the next generation.
- If an empty cell is surrounded in the present generation by exactly three neighbors, the cell will be on (ie: born) in the next generation.
- If a cell has no neighboring live cells, or only one neighbor, it dies of loneliness and will be turned off in the next generation.
- If a cell has four or more live cells neighboring it, it will die in the next generation from overcrowding.

These rules are to be applied simultaneously to every cell in the pattern. The application of the rules to every bit in the field constitutes a generation. See figure 1 for an example of rule applications.

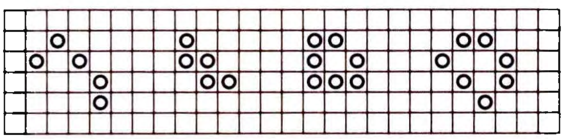


Figure 1: Transformation of a Life pattern through three generations. This process is sometimes referred to as the automation of the pattern. Generation 0 is the original pattern of live cells. The succeeding generations proceed according to the rules of birth, existence and death.

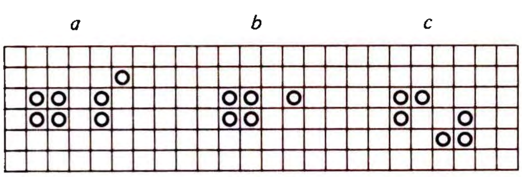


Figure 2: Example of object contiguity. Group a is considered to be two distinct objects; groups b and c are considered to be single objects.

Unresolved Questions

What is a unique object in this universe of cells? What is a collection of objects? How do we tell them apart? These are difficult questions to answer conclusively. For the purposes of this article, an object is a cluster of connected bits or cells, a collection of clusters which will cause births by being near one another, or a collection of clusters that prevent some birth that would otherwise occur. Figure 2 gives some examples of patterns that would be objects and some that would not.

A collection of distinct objects is referred to as a *constellation*. Some constellations are so common that they are named as though they were a single object. Some of these are presented in figure 3.

Objects

Most people with access to some sort of computer have probably had a chance to observe the variety of patterns that exist within Life and to note some of the special properties particular to some of these objects. In order to be able to manipulate these objects, they have been classified.

The major groupings of classification are still lifes, oscillators, spaceships, uniform propagators, and a catch-all group of random objects. A rough outline of this system is shown in table 1. I shall attempt to describe

	Subclass	Number of objects known	Smallest object(s)
Class I (still lifes)	subdivided by symmetry	∞	block
Class II (oscillators)	(IIa1) flip flops (IIa2) on-offs	∞	blinker beacon
	(IIb) billiard table configurations	>100	MIT oscillator
	(IIc) inductors	4	tumbler
	(IId) pulsators	8	mazing, pentadecathlon
	(IIe) shuttles	5	shuttle
	(IIf) eater bound	23	two eaters
Class III (spaceships)	(IIIa) diagonal	1	glider
	(IIIb) orthogonal	3	lightweight spaceship (LWSS)
Class IV (uniform propagators)	(IVa) stationary	2	glider gun
	(IVb) moving (puffer trains)	4 types	switch engine
Class V (random)	subdivided by type of objects in census	∞	bit (single cell)

Table 1: Classes and subclasses of objects occurring in Life, along with supplementary information.

Number of live cells	Number of still life patterns
1	0
2	0
3	0
4	2
5	1
6	5
7	4
8	9
9	10
10	25
11	46
12	121
13	240
14	619

Table 2: The number of small still life patterns which occur for each number of live cells up to 14.

a *b* *c* *d*

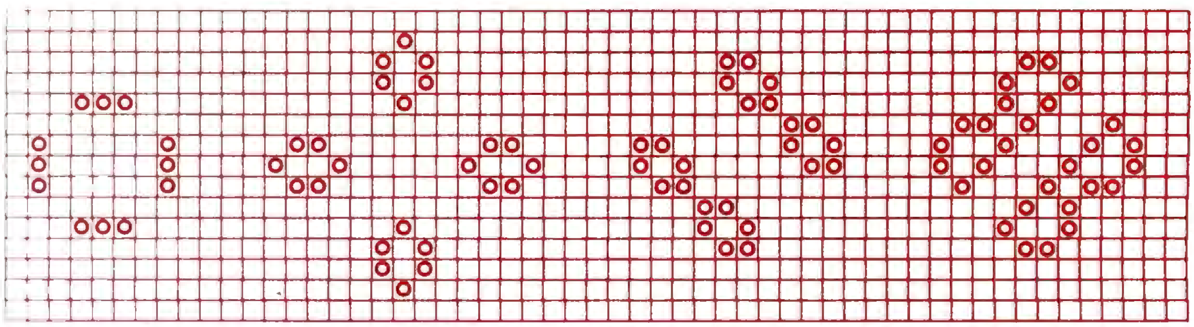
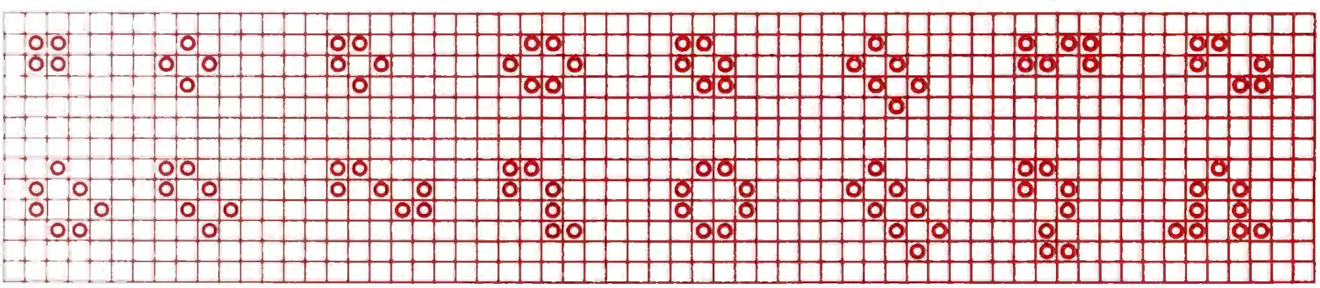


Figure 3: Commonly occurring constellations. These are not a single object, but bear names for convenience, as follows: a, traffic light; b, honeyfarm; c, fleet; d, bakery.

a *b* *c* *d* *e* *f* *g* *h*



i *j* *k* *l* *m* *n* *o* *p*

Figure 4: An assortment of still life objects. These remain stable from generation to generation when not disturbed by other objects. They bear names as follows: (top row, left to right) a, block; b, tub; c, boat; d, beehive; e, ship; f, barge; g, snake; h, aircraft carrier; (bottom row, left to right) i, burloaf; j, long boat; k, long snake; l, period 3 eater; m, pond; n, long barge; o, shillelagh; p, hat.

each class and some of the objects of particular note within each class.

Class I: Still Lifes

Still lifes are objects in which there are no births or deaths and so remain the same from generation to generation. These particular objects are fairly easy to enumerate. An associate of mine, Peter Raynham, wrote a program which found all still lifes of less than 15 bits. The statistics of their distribution are presented in table 2. Some of the smaller ones are shown in figure 4.

One of the most practical uses of a still life is as an *eater*. An eater is an object capable of destroying or modifying another object and being able to return to its original configuration. Still lifes are good for this since they are able to attack any configuration at any phase (they are period 1 objects and do not change).

At present we know of three different eaters, each able to attack different types of objects. By differing objects, I mean objects that have different border configurations. Since the eater attacks only the outside surface of an object, this outer surface determines which type of eater might be suitable for use. Each eater will be described with an object that it can "eat" to show how that eater works.

The smallest member of the eater family is the *block*, shown in figure 4a. The block is effective in consuming objects that have one connected bit in the row facing it (as in figure 5). Its other reason for utility is the fact that it is very small. In oscillators and spaceship guns, where there may be little room for the removal of spurious debris, there is usually enough room for a block.

Note:

A bibliography containing all *Scientific American* Life articles referred to in text is provided at the end of this article.

The second object of this family is the most important and versatile. It is the 7 bit still life referred to in figure 4l as the period 3 eater. This object attacks other objects that exhibit a flat connected outer border that is at least two bits long. Figure 6 shows such an attack. Almost all objects will develop this type of border if they expand. This property renders the period 3 eater invaluable. Although it is not quite as small as the block, it is still very much smaller than any other of the eater family.

The third such object, the period 6 eater, exhibits similarities to the period 3 eater in the way it eats; however it requires six generations to dispose of its prey and return to its initial state, whereas the previous eater takes three generations. This increase in time is important for success if the object being eaten has left some transient debris near the eater. If the eating mechanism were to reform itself quickly, this debris could kill the eater. In this case, the eater does not reform for an extra three generations, during which time this debris may well vanish.

Most of the period 6 eater's prey are the same as the period 3 eater's; but both are able to attack certain additional objects, complementing each other very nicely. Figure 7 shows the period 6 eater conveniently disposing of a block.

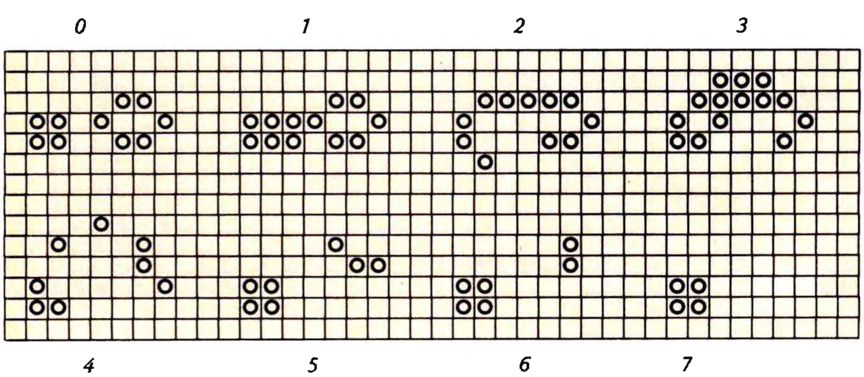


Figure 5: A block devours a beehive. This process requires seven generations, as shown here.

Class II: Oscillators

Oscillators are nonmoving objects with periods of two and greater. A *blinker*, shown in figure 8a, is a simple oscillator consisting of three cells alternating in subsequent generations between a vertical and horizontal row. At present, we know of roughly 150 unique oscillators with a period greater than two.

There is a large undetermined number of period 2 oscillators, since they are very easy to construct. The oscillators have been subclassified by relating their mechanisms and their degree of naturalness. (Natural objects are those which may evolve from random patterns of live cells without intervention by the experimenter.)

Since there are only two basic ways in which a period 2 oscillator can work, these objects are very well defined. Therefore, they are assigned to their own subclass (class IIa). They must work as *flip flops*, *on-offs* or a combination of the two. In a flip flop, deaths occur because of *under*population. In an on-off, any deaths that occur are due to *over*population. (This is almost always true. Figure 9 is a period 2 oscillator, which if traced, will reveal that it adheres to both definitions.) A variety of small period 2 oscillators is shown in figure 8; the type of each oscillator is also given.

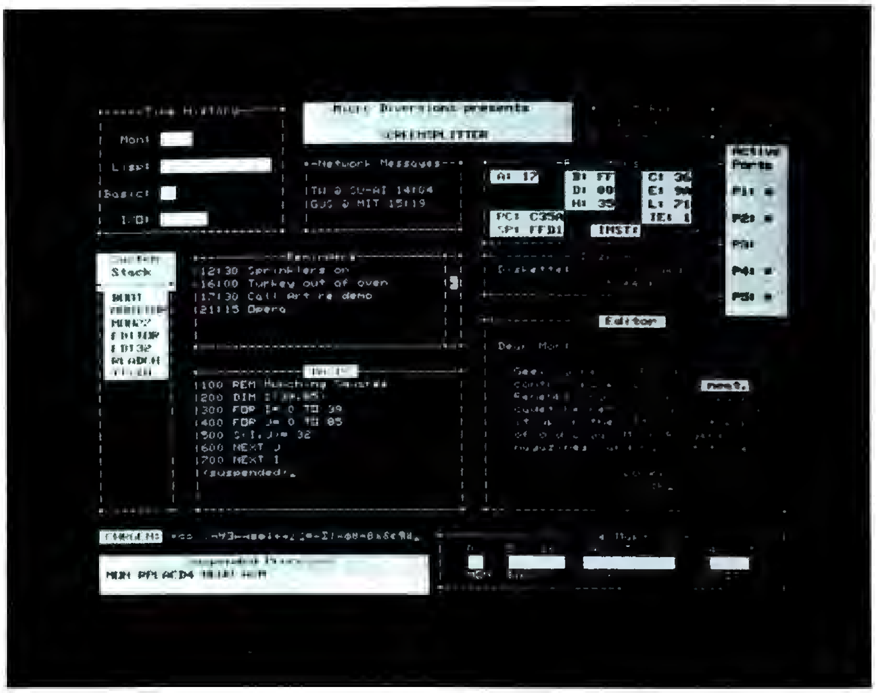
Next in the hierarchy are *billiard table*

See It All at Once!

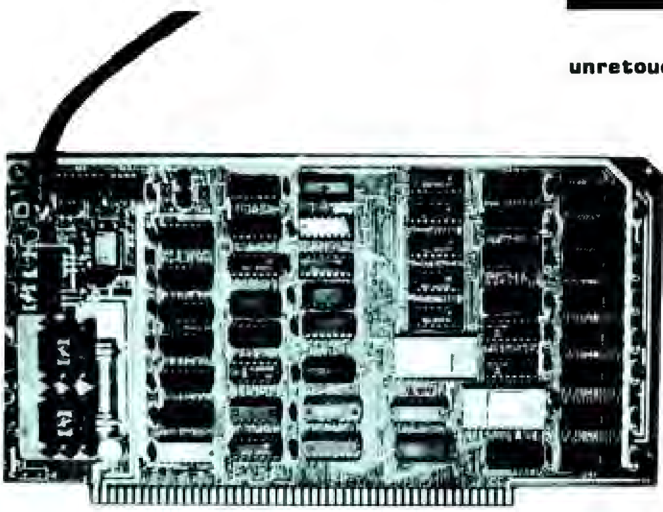
h a **Screensplitter**

Text Editing and Multiple Process Display System

from Micro Diversions)



unretouched photo   13" or larger slow phosphor monitor recommended



S-100 Buss

- 40 lines, 86 characters/line UPPER/lower case full ASCII with special symbols
- Onboard intelligence: 1K byte fast subscreen control, positioning and text formatting software
- Custom reprogrammable character generator for unusual applications
- 140 page Owner's Guide covering hardware, software, system integration
- Available from stock

$295 Kit
$395 Assembled
$15 Owner's Guide only

VISA
master charge

Micro Diversions, Inc.
8455-D Tyco Road
Vienna, Virginia 22180
(703) 827-0888

Circle 225 on inquiry card.

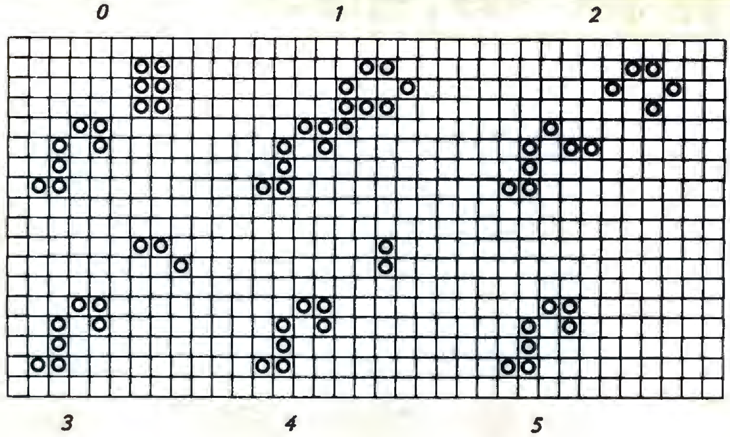


Figure 6: The most versatile eater object, the period 3 eater, devours the precursor pattern to beehive. While in isolatioı a 7 cell still life, this eate attacks other objects witı a flat connected outeı border at least two cells in length.

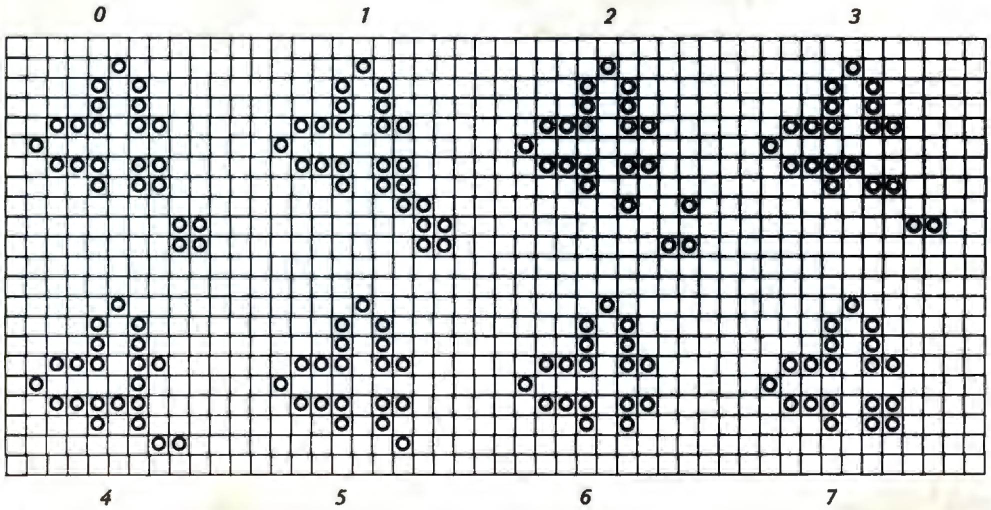


Figure 7: A period 6 eater attacks and eats a block. This new eater is notably more symmetrical than the period 3 eater. Functionally, each complements the capability of the other.

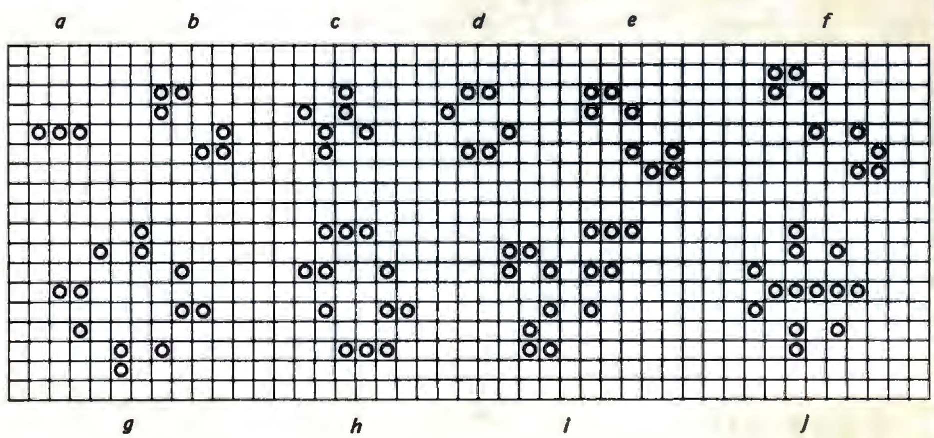


Figure 8: A variety of small period 2 oscillators. These objects alternate in succeeding generations between two patterns. Those on the top row bear names as follows (left to right): a, blinker; b, beacon; c, clock; d, toad; e, bipole; f, tripole; those on the bottom row are unnamed flip flops. The beacon is the only example here of an on-off type oscillator.

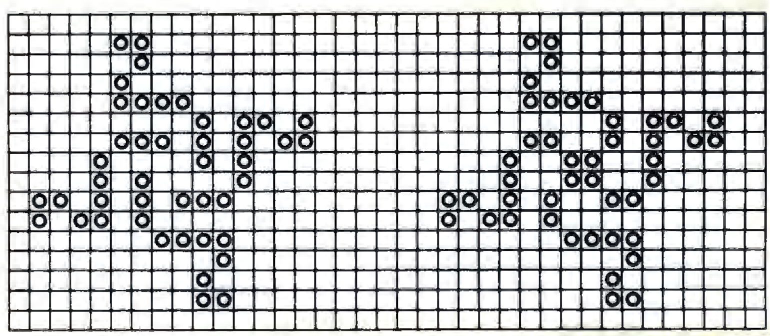
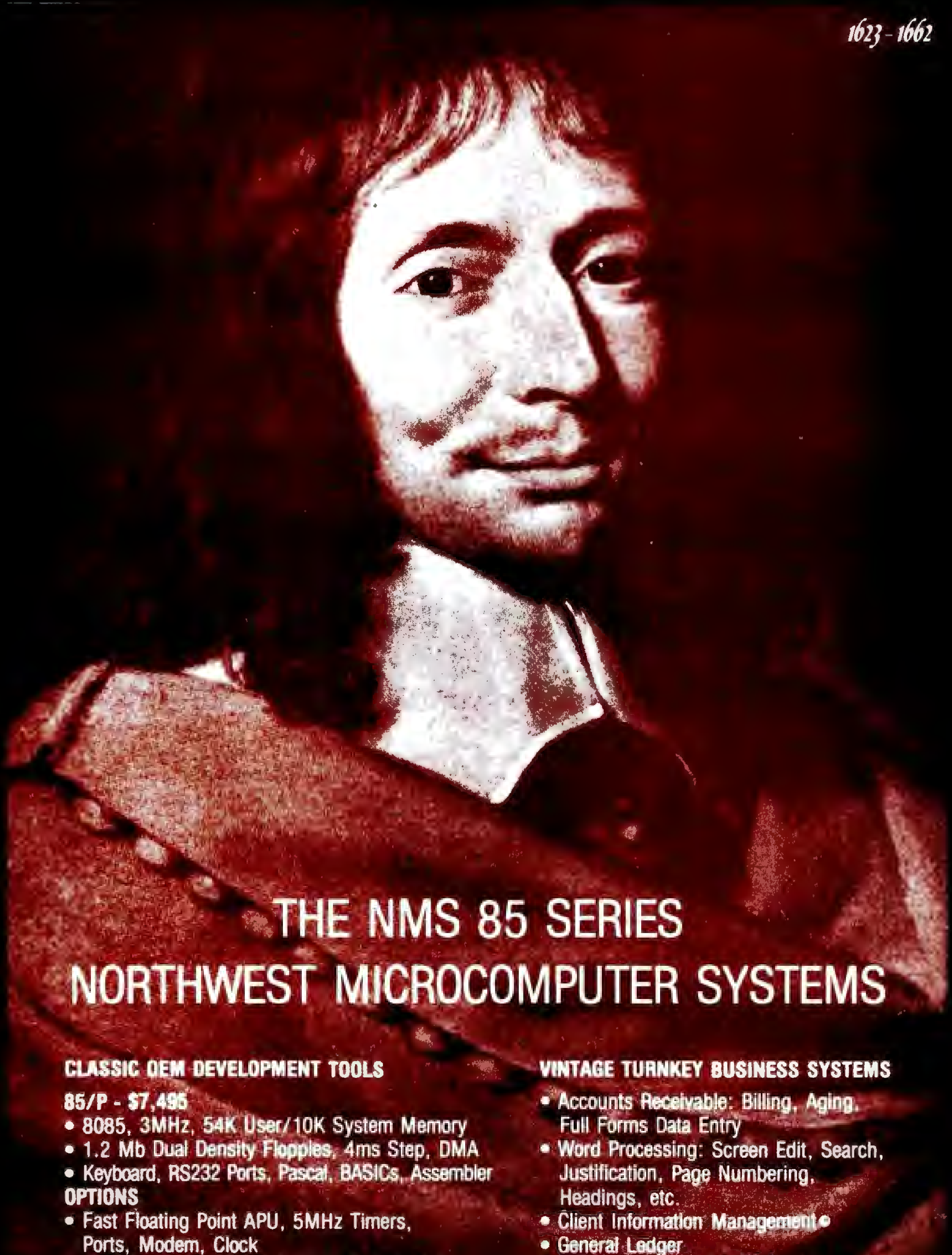


Figure 9: A period 2 oscillator which functions both as a flip flop and as an on-off.

Blaise Pascal



121 EAST ELEVENTH, EUGENE, OREGON 97401 (503) 485-0626

Circle 286 on inquiry card.

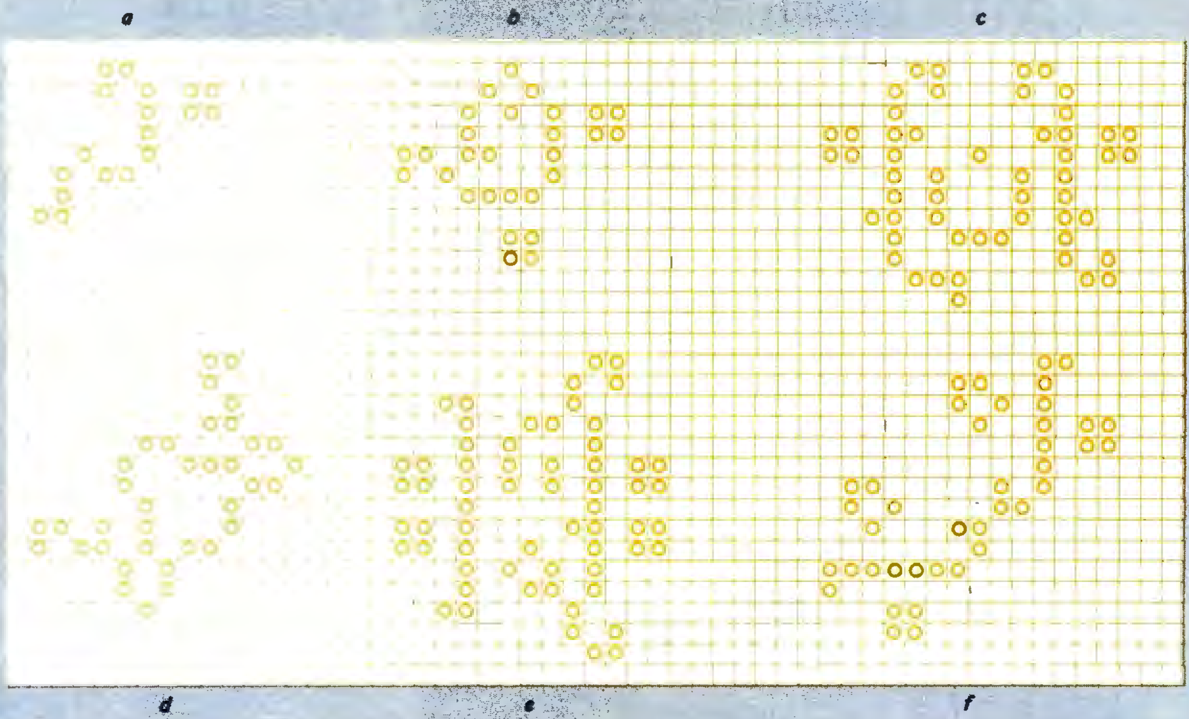


Figure 10: Billiard table configurations. These oscillate within an enclosed area, as do balls on a billiard table. These are artificial objects which have not occurred unless specifically constructed by the experimenter. They tend to be large; the smallest is composed of 18 live cells. Those illustrated bear names as follows: a, MIT oscillator (a period 3 object); b, burloaferimeter (period 7); c, an unnamed period 8 object; d, wavefront (period 4); e, an unnamed period 5 oscillator; f, an unnamed period 9 object.

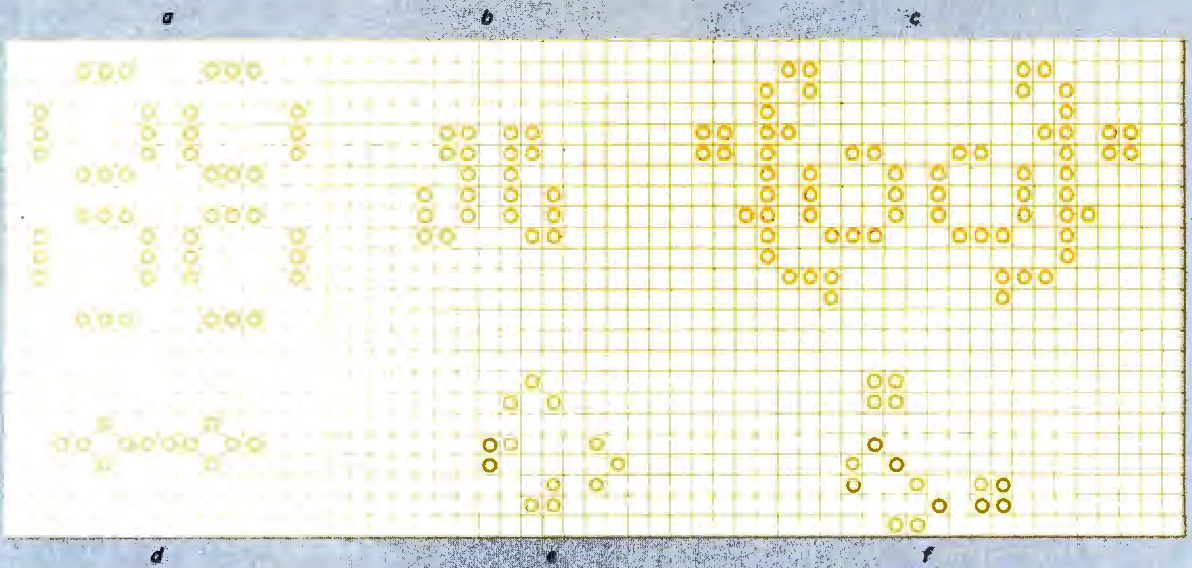


Figure 11: Inductor and pulsator oscillators. These are natural objects which may appear from automation of random patterns. Inductors possess an imaginary line of symmetry which pulsators lack. They are called by the following names: a, pulsar (an inductor of period 3); b, tumbler (period 14 inductor); c, an unnamed period 8 inductor; d, pentadecathlon (pulsator of period 15); e, mazing (period 4 pulsator); f, unix (period 6 pulsator). The pentadecathlon is of particular historical significance.

introducing . . .

YOUR NEW LIFE STYLE!

Discover How to Improve Your Personal Abilities

An amazingly powerful application of a computerized personal information management system. In handy booklet form.

- Increase your personal capabilities
- Save money
- Improve your ability to plan
- Locate important facts quickly
- Eliminate the drudgery of routine chores

These all add up to a better life style for you! A new way of living which can be more pleasant, bring you more happiness and success.

Information. That's the key. Your command of information is what gives you the power to succeed.

Information For What?

Information you need to make decisions . . . to solve problems . . . to seek creative solutions in real life situations. Information that when under your control will amplify your personal abilities.

You've Heard of the Information Explosion?

Now it's time for an explosion in your personal capacity to deal with a wealth of information. The personal computer is the answer. And now the means is available for you to use it to your personal advantage.

Can a Computer Really Do It?

You bet it can! The power of a personal computer is fantastic. Untold amounts of information can be processed in the blinking of an eye. Chores you've found laborious can be done by your simple command. Facts that have taken hours to find can now be located at the snap of your finger. That's what a personal computer can do . . .

. . . if YOU know how!

YOU? Yes, you! You need to have the know-how to unleash the power of the microcomputer for your benefit.

Until now only the programmers and designers could make such profitable use of the personal computer. PIMS – the Personal Information Management System – has changed that. You don't have to be a programmer to use PIMS. PIMS is a program which you type into a personal computer. Then it's just a matter of following the instructions for setting up your own tasks for computer assistance.

Can It Really Be That Easy?

Sure. That's the beauty of a program for information management. It's an alternative to programming from your standpoint. All you do is define the job you want the microcomputer to do. Express yourself in simple commands and statements. Then the microcomputer plus PIMS does the rest.

What Can It Do?

PIMS will enable you to function with increased effectiveness. Use it to balance your check book. It can keep an inventory of your possessions. (Ever store something and afterwards forget where? PIMS will tell you the location!) Use PIMS to plan your day, vacation, education and important life events. PIMS can tutor math, keep your personal mailing list or telephone directory. Use it to keep track of personal disbursements. It can even be used to schedule your TV viewing. And more. Much more. There's practically no limit to the type of applications.

Here's what you get.

All in one neatly bound volume,

- The microcomputer and its potential for personal use
- Ways in which your microcomputer can be helpful in everyday life and serving basic personal needs
- PIMS – Personal Information Management System. All the basics and jargon
- PIMS program outline and flow chart
- 15 sample ways in which PIMS can work to your personal advantage
- The complete listing of a practical functional program. Written in BASIC language. Ready to use!

The PIMS program was prepared by SCELBI using Microsoft† compatible BASIC as used in a wide variety of personal computers. Systems like Apple II, Radio Shack TRS-80 (level II), Ohio Scientific 400 and many other small computer systems.

Don't Have a Computer?

Even if you are still just thinking about getting your own personal computer, you should have this book. It will show you in clear understandable terms what you'll be able to accomplish with your own microcomputer.

Coming Soon!

PIMS – Personal Information Manage-System will soon be available in handy booklet form. First-off-the-press copies will be ready for shipment in January. At only $9.95 (plus shipping and handling charges) it's got to be the greatest bargain to come along!

An Offer You Won't Want to Refuse:

Now, as a pre-publication offer, you can get your copy for $1 off. That's right, if you act now, before January 1, you can get your copy of PIMS for just $8.95.*

Don't Wait.

Get your copy now! You'll discover how you can find your new life style – with PIMS.

*Prices shown are for North American customers. Pricing, specifications, availability subject to change without notice. Personal checks may delay shipment. IMPORTANT! Include 75¢ postage/handling for each item to be delivered by U.S. Mail Book Rate, or $2 for each item to be shipped First Class or UPS.

† A trade mark of Microsoft, Inc.

SCELBI Publications
P.O. Box 133 PP STN
Milford, CT 06460

YES! I want to take advantage of the PIMS pre-publication offer. Enclosed is $8.95 plus 75¢ postage/handling. Or, better yet, please charge my Master Charge or VISA account.

Card No.__________

Bank No.__________

Name__________

Address__________

City__________

State__________ Zip__________

Signature__________

configurations (class IIb). These oscillators are configurations that oscillate within an enclosed area, like balls on a billiard table. Billiard table configurations are considered to be very artificial, since they have not turned up in the histories of any random objects. By this, I mean that if live cells are placed randomly on the plane, the patterns which they generate probably will never evolve into an artificial object, such as a billiard table configuration.

They are quite large, as evidenced by the examples in figure 10. The first example is the smallest such object, and it consists of 18 bits. This subclass of oscillators contains the only known examples of oscillators with periods of 7, 10 and 11.

The next class, *inductors* (class 11c), are natural oscillators that exist in two or four pieces with an imaginary line of symmetry between them (exhibiting one-way or two-way orthogonal symmetry). *Pulsators* (class IId) are also so far considered to be natural oscillators except that they do not have this line of symmetry. One of their properties is that they require no external stimulus to continue oscillating.

The aforementioned subclasses have greatly similar characteristics, so I have grouped them together. Most of the initial oscillators that were found were from this group, since the methods for harnessing random objects into oscillators were not known at the time.

Some of these oscillators are presented in figure 11; the most important of these is the *pentadecathlon*. This object throws off several sparks (small collections of dying bits) that can be used to reflect a glider, reflect two gliders, turn a glider into a block, turn a block into a glider, etc. Some of the early research into Life probably might not have occurred if this object had not been discovered.

Shuttles (class IIe) are important for the existence of much of the interesting research into Life. Shuttles are objects that move back and forth with a relatively large period. The two primary shuttles, the *basic shuttle* and the more complex *twin bees*, leave debris at their extremities which would fatally wound the shuttles if the debris were not removed before they returned (see figure 12). This is one of the uses of the eaters that was discussed in the section on still lifes. In the examples I have used blocks to remove the debris from the ends, but just about any of the eaters would have suited some phase of this debris. The debris left behind may at first seem to be somewhat of a bother, but without it there would most likely not be any known glider guns (defined later).

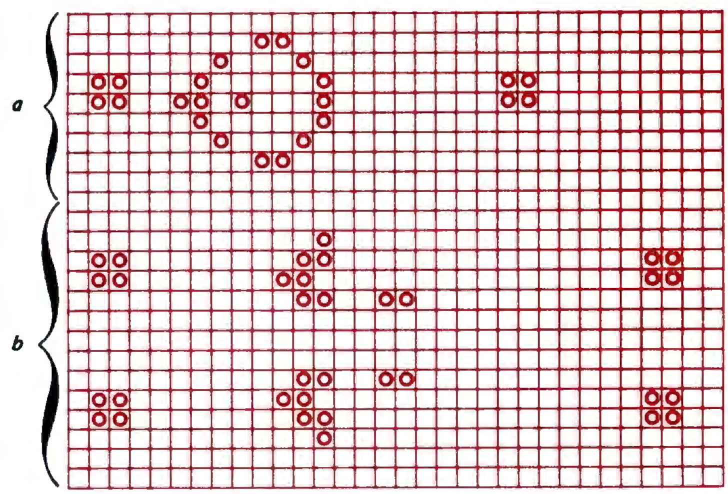


Figure 12: Shuttle objects. Object a is the basic shuttle; object b is the twin bees shuttle. These move back and forth with a relatively long period. Eaters are used to remove debris from their path.

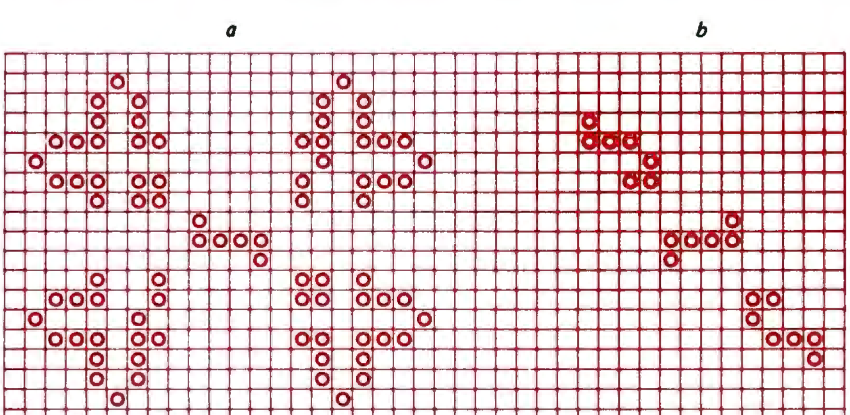
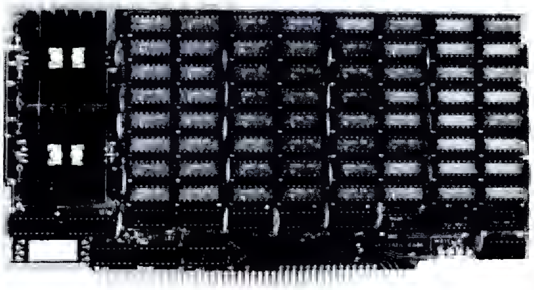


Figure 13: Two eater bound oscillators. These differ in that they are stabilized by two different eaters. Oscillator a has a period of 6; oscillator b has a period of 5.

The very last class (class IIg) contains *eater bound oscillators*. These oscillators consist of patterns which generally must be manipulated in order for the object to return to its initial state. In figure 13 a good example of two eater bound oscillators is given that also shows the differentiation between two eaters acting on the same object (which is not often possible). A period 52 oscillator (figure 14) is shown to illustrate the unusual properties of objects being eaten. The center object will be attacked by one eater twice each time it rotates (the object rotates 90° every 13 generations). The example in figure 15 is a period 6 oscillator using the period 6 eater. The 7 bit eater is not suitable here because it would have returned to its original state too soon and would have attacked the reforming object. (If the 7 bit eater

Wondering which memory is best for you?

base 2• offers the following products to the S-100 market at the industry's lowest prices:

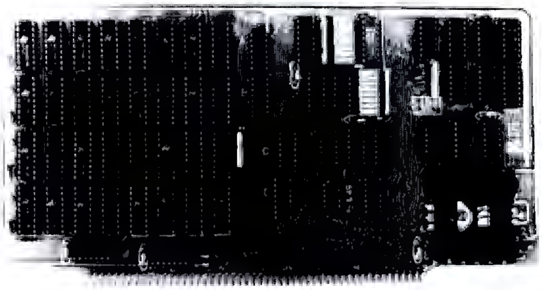


8K Static Memory Board

This 8K board is available in two versions. The 8KS-B operates at 450ns for use with 8080 and 8080A microprocessor systems and Z-80 systems operating at 2MHz. The 8KS-Z operates at 250ns and is suitable for use with Z-80 systems operating at 4MHz. Both kits feature factory fresh 2102's (low power on 8KS-B) and includes sockets for all IC's. Support logic is low power Schottky to minimize power consumption. Address and data lines are fully buffered and 4K bank addressing is DIP switch selectable. Memory Protect/Unprotect, selectable wait states and battery backup are also designed into the board. Circuit boards are solder masked and silk-screened for ease of construction. These kits are the best memory value on the market! Available from stock . . . **8KS-B** **$125** (assembled and tested add $25.00)
8KS-Z **$145** (assembled and tested add $25.00)

16K Static Memory Board

Base 2 can now offer the same price/performance in a 16K static RAM as in its popular 8K RAM. This kit includes 8K bank addressing with 4K boundary address setting on DIP switches. This low power unit provides on-board bank selection for unlimited expansion . . . No MUX board required. Using highest quality boards and components we expect this kit to be one of the most popular units on the market. Available in two speed ranges, the 16KS-B operates at 450ns while the 16KS-Z operates at 250ns.
16KS-B **$285** (assembled and tested add $25.00)
16KS-Z **$325** (assembled and tested add $25.00)

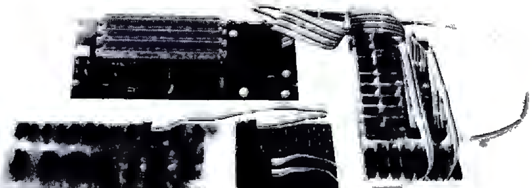


Z-80 CPU Board

Our Z-80 card is also offered in two speed ranges. The CPZ-1 operates at 2MHz and the CPZ-2 operates at 4MHz. These cards offer the maximum in versatility at unbelievably low cost. A socket is included on the board for a 2708 EPROM which is addressable to any 4K boundary above 32K. The power-on jump feature can be selected to address any 4K boundary above 32K or the on-board 2708. An On-board run-stop flip-flop and optional generation of Memory Write allows the board to run with or without a front panel. The board can be selected to run in either the 8080 mode, to take advantage of existing software, or in the Z-80 mode for maximum efficiency. For use in existing systems, a wait state may be added to the M1 cycle, Memory request cycle, on-board ROM cycle, input cycle and output cycle. DMA grant tri-states all signals from the processor board. All this and more on top quality PC boards, fully socketed with fresh IC's. **CPZ-1** **$110** **CPZ-2** **$125**

S-100 for Digital Group Systems

This kit offers, at long last, the ability to take advantage of S-100 products within your existing Digital Group mainframe. Once installed, up to four S-100 boards can be used in addition to the existing boards in the D.G. system. The system includes an "intelligent" mother board, ribbon cables to link existing D.G. CPU to the DGS-100 board and a power wiring harness. The DGS-100 is designed to fit in the 5-3/4" x 12" empty area in the standard D.G. cabinet. It may seem expensive but there's a lot here! End your frustration! **DGS-100** **$295**



base 2. inc.

Send for more details on these products. Get on our mailing list for information on more soon to be announced products at factory-direct prices from BASE 2. Why pay more when you can get the best at these prices???

P.O. Box 3548 • Fullerton, Calif. 92634
(714) 992-4344
CA residents add 6% tax
MC/BAC accepted • FOB — U.S. destination

is used, the patterns results in two blinkers, six blocks, and one tub in 110 generations.)

Class III: Spaceships

Regrettably, there have been no new spaceships reported since the orthogonal spaceships presented in *Scientific American* in 1971. These are summarized in figure 16. The *glider* (figure 16a), which features diagonal movement, has been used for many simulations and constructions.

Movement by an object of one space in one generation is referred to as movement at the speed of light (c). There is no distinction made between diagonal and orthogonal movement, even though algebraically the distances are not the same. The glider travels at c/4 and the three other spaceships travel at c/2. The interesting thing to note is that the three larger spaceships travel orthogonally. The orthogonal spaceships are most useful in several of the types of puffer trains to be discussed in the next section.

Class IV: Pattern Producing Mechanisms

Class IV is divided into two sections: the first contains static patterns that produce moving progeny, and the second contains moving patterns that produce some type of stationary or moving output.

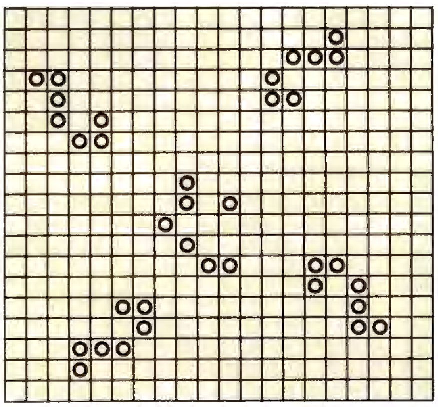


Figure 14: A long period eater bound oscillator. This object has a period of 52; 13 generations are required for 90° of rotation. The central section is attacked twice by one eater each time it rotates.

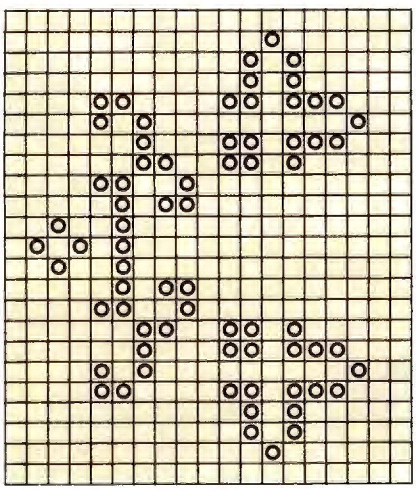


Figure 15: A period 6 oscillator which employs the period 6 eater. This matching of period frequencies prevents the eater from disrupting the re-forming central group.

Spaceship Guns

Class IVa consists of *spaceship guns*. These objects eject projectiles of class III objects. The main two objects of class IVa are the *glider guns* of primary period 30 and 46. There are no primary guns which produce any of the other three spaceships. However, such a mechanism can be built using glider guns.

The period 30 glider gun (figure 17) works by having two shuttles of the type presented earlier aimed at one another. The debris that would normally be removed by eaters collides and just happens to create a glider that escapes without harming the shuttles. The period 30 glider gun is of paramount importance to simulations in Life and the possible existence of computing mechanisms. These implications will be discussed in a later article.

The period 46 gun, known as a *newgun*, also works by having two shuttles collide. It may be seen in figure 18. In this case the shuttle consists of two B heptominos (described later) travelling opposite one another to produce debris at both ends of travel. A glider is produced when these two shuttles, which are of the twin bees type, collide at right angles. There are other arrangements of this shuttle that produce gliders in other ways, including an ambidextrous variety.

There is another interesting variation: if one of the debris removing blocks is removed from the end of one of the twin bee shuttles, the gun will still work.

Puffer Trains

Puffer trains are patterns that move and leave debris in their wake. Because these patterns do move, as opposed to the stationary spaceship guns, they are not only able to produce moving debris but also trails consisting of stationary objects. Leaving stable objects is useful when the intention is to produce a train of puffers to build some sort of construction on the fly.

The three basic puffer trains all work by different means. The train which was discovered first is presented in figure 19. The center object is a *pre-B heptomino*, which, if traced, will seem to move forward until the debris in the back of it stops the uniform forward motion. In this case, the B heptomino is bounded by two lightweight spaceships able to control the object; the whole configuration puffs along at c/2.

This object reaches a stable period of 140 after a startup time of over 1000 generations. Additional spaceships may be added to the end of the object to further adjust the output from it in order to reduce the final period, the startup time necessary to reach a stable period and to adjust the output to blocks, gliders, etc.

A type of puffer similar to the previous one is called a *Schick ship* (after its discoverer). This is an interesting object (consult figure 20) in that the "engine" is really a tagalong, an object capable of being pulled along behind another object (usually a spaceship). Here, a heptomino follows a pair of mirrored spaceships. It is quite remarkable that this configuration leaves a small trail of debris behind it and that, although this debris would die if left alone, additional spaceships following behind are able to trigger the debris into varying forms of static debris. The static debris can be left behind and used later. It is relatively useful for building armadas because of the relative simplicity of creating this object from gliders (producing a basic ship requires 11 gliders).

The last type of puffer train is the smallest, a mere 11 bits at startup — the size is somewhat larger when the final repeat cycle is known, since there is transient debris in the field. This particular train travels very slowly, taking 96 generations to traverse eight spaces (speed c/12). It is also very unusual in that it is the only known puffer train that travels diagonally — the same direction as the glider, but three times as slow.

Unlike the other puffer engines, this train does not require that any other spaceship exist to bound it. To stabilize the basic engine, a block must be placed somewhere in the debris produced by the object to prevent the debris from destroying it. If the engine is run without a stabilizing block, some rising debris finally catches the engine after 11 full cycles and destroys it. The remaining field settles down to a final census only after 3911 generations!

Pertinent to the above paragraph is the fact that random patterns are quite often able to produce certain types of edge configurations, which enable them to surge forward with a great burst of speed for short periods of time. In the case of the *switch engine*, when some random exhaust manages this type of movement, this slow moving engine is easily caught.

The switch engine (presented in figure 21) will produce, after its startup time, eight blocks every 288 generations. Other debris can be produced, including gliders. Since this train travels so slowly, there are presently no real uses for it.

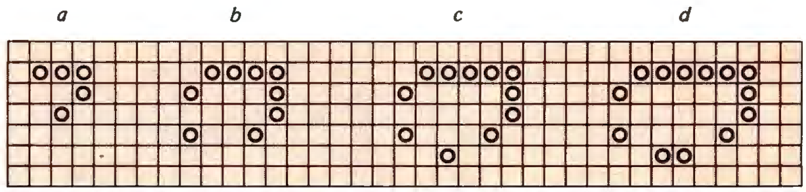


Figure 16: The four known spaceships which occur in Life. Their appellations are: a, glider; b, lightweight spaceship; c, middleweight spaceship; and d, heavyweight spaceship. The glider travels diagonally at a rate of one space every four generations. The other three travel orthogonally at one space per two generations.

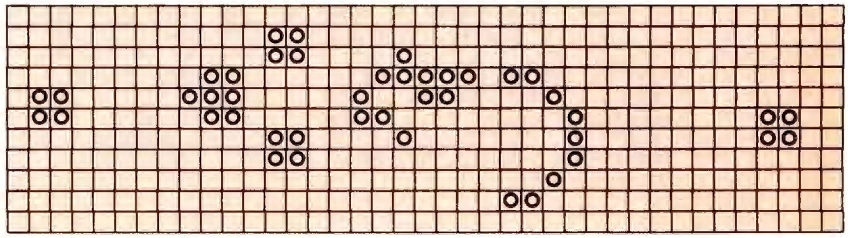


Figure 17: This glider gun, which has a period of 30 generations, was the first object of class IVa to be discovered. It periodically emits a glider which travels away diagonally. The four block still lifes are used as eaters to dispose of debris. Glider guns are of great importance in simulations, where gliders are made to collide, thus forming new objects.

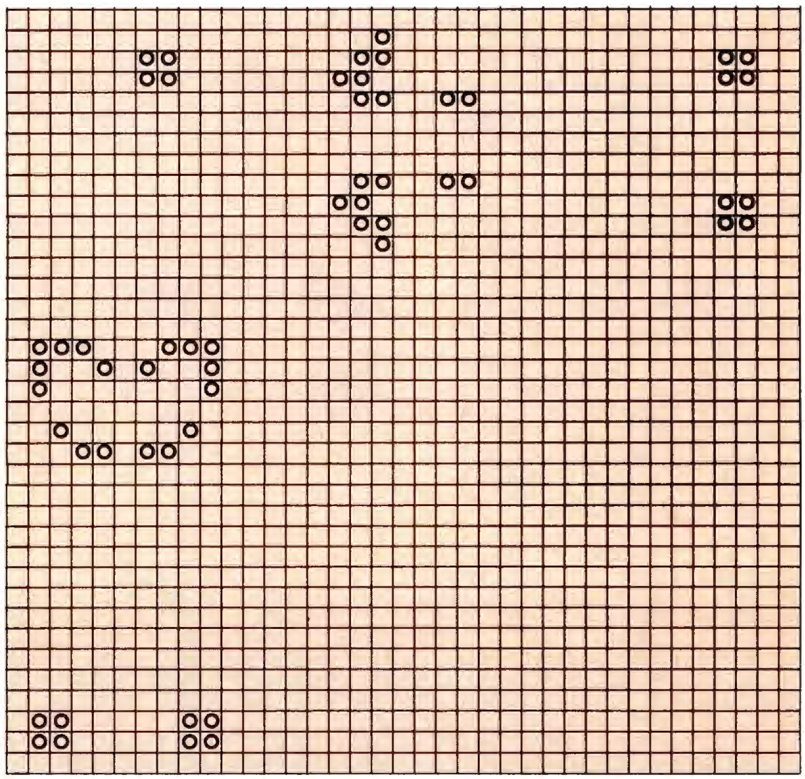


Figure 18: A period 46 glider gun which is called the newgun. Two twin bees shuttles collide at right angles to produce one glider every 46 generations. As before, the block still lifes are used to remove debris which could cause disruption of the formation.

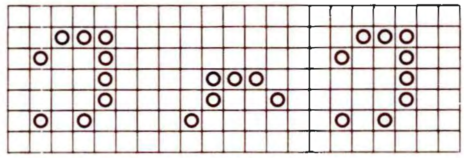
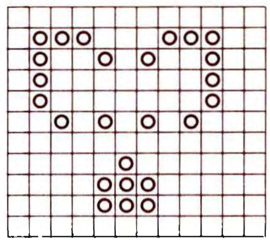


Figure 19: A puffer train constructed from the precursor to a B heptomino and two lightweight spaceships. After a startup period of over 1000 generations, it stabilizes into a period of 140 generations.

Figure 20: A Schick ship puffer train. The engine in this object is pulled along behind another object. In this example, a heptomino tags along behind a pair of mirrored spaceships. This object has a period of 12.



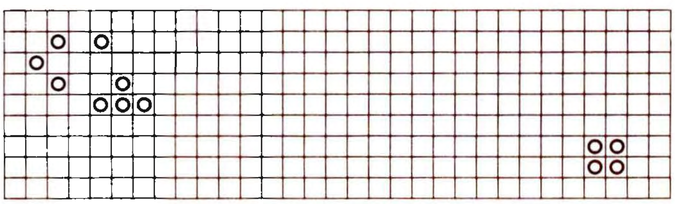
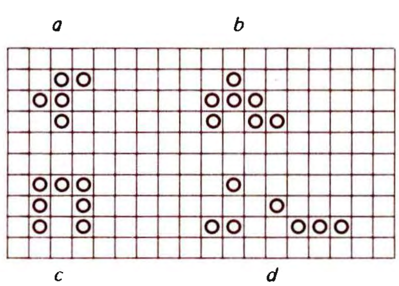


Figure 21: The switch engine puffer train, with a basic period of 96. After startup, it produces eight blocks every 288 generations after the initial stabilization of debris. Although much various debris may be created with this train, its slow speed limits its usefulness.

Figure 22: Several common nonterminal random objects. These are designated as follows: a, R pentomino; b, B heptomino; c, π heptomino; and d, acorn. The acorn is a Methuselah type object. Finding all the descendants of such an object is a difficult challenge.



Class V: Random Objects

A *random object* is simply anything that does not fit in any of the above classes. It appears that all random objects eventually become something from one of the above classes. It has been assumed that there are no objects that expand *irregularly* forever (this is a common problem in other cellular spaces using other rules). There are some very popular *nonterminals* in life, which, due to their commonality, have been given names. In some cases these have been rather heavily investigated. In figure 22 are some of the more common nonterminals and their names.

The most common object of this class must be the oft publicized *R pentomino* (figure 22a), which many people still believe runs forever. The result of this pattern was, however, published in *Scientific American*; it runs for 1103 generations, producing four blinkers, eight blocks, one boat, four beehives, one ship, one burloaf, and six gliders.

The *B heptomino* (figure 22b), with a census of three blocks, one ship and two gliders in 148 generations, is one of the more heavily investigated objects, as is evidenced by some of the material presented in this article. It has the following interesting property: the front configuration of the object moves along to reappear the same every other generation, but flipped over.

A close relative of the previous object is the *π heptomino* (figure 22c) with a census of five blinkers, six blocks and two ponds in 173 generations. Phase 1 of this object was called a *blasting cap* by the artificial intelligence researchers at Massachusetts Institute of Technology (MIT); we call phase 3 a *house*. If you trace the house for 30 generations, you will notice that a house reappears at the front of the debris ten spaces ahead of where it started. The house does not appear again after this because the debris catches up with it and kills it. Many attempts have been made to stabilize this object, with no success as yet.

A random object that consists of fewer than ten bits and that has descendants enduring for more than 50 generations is referred to as a *Methuselah*. The acorn pattern (figure 22d) is presently the record holder for duration. This is presented as a challenge to anyone who would like a difficult object to trace.

We hope that some of our investigations into the more exotic corners of Life will inspire readers to try their hands at this fascinating pastime.■

About the Author

David Buckingham is an undergraduate science student at the University of Waterloo. He made a number of contributions to the now defunct publication Lifeline. *Most of these had to do with oscillators, which constitute his main field of interest in Life. He has at present found over 100 oscillators of periods greater than two.*

Buckingham's most productive area of research has been the devising of glider collisions to produce objects of classes I thru IV. As of August 1978, he has managed to create collisions to produce all of the presently known 1105 objects of less than 15 bits in size.

Early issues of BYTE *carried a never completed series by Carl Helmers inadvertently entitled "LIFE Line," which was also the name used for Robert Wainwright's newsletter. These Helmers articles appeared as follows:*

"LIFE Line 1," September 1975 BYTE, volume 1, number 1, pages 72 thru 80;
"LIFE Line 2," October 1975 BYTE, volume 1, number 2, pages 34 thru 42;
"LIFE Line 3," December 1975 BYTE, volume 1, number 4, pages 48 thru 55;
"LIFE Line 4," January 1976 BYTE, volume 1, number 5, pages 32 thru 41.

A bibliography of Scientific American *information on LIFE (all references are to Martin Gardner's "Mathematical Games" column).*

October 1970: page 120. This is the original Life article, including the definition of the facts of Life, and illustration of numerous fundamental patterns.

November 1970: page 118. Answers to several questions posed in the first article on the subject, including definition of the several varieties of spaceships.

January 1971: pages 105, 106 and 108. Continued progress on the Life front including answers to several unsolved questions and results of a flurry of computer Life activity.

February 1971: Special "Mathematical Games" article on "cellular automata theory."

March 1971: pages 108 and 109. Short note about progress made by John Conway and R William Gosper, plus illustration of a large scale flip flop pattern which is delicately balanced and easily destroyed by minor disturbances such as impact of a glider.

April 1971: pages 116 and 117. Examples of fuses, the 5 cell cross series, and announcement of Robert T Wainright's *Lifeline* newsletter.

November 1971: page 120. Short note on continued progress at the MIT AI Laboratory.

January 1972: page 107. The discovery of the eater by Bill Gosper at MIT.

Hurry down to your public library if you wish to use these references, as due to lack of space some local libraries may be committing the unspeakable crime of throwing away Scientific American . . . .**CH**

The January 1979 issue of BYTE will contain an article by Mark Niemiec which describes several algorithms for Life. Readers who wish to experiment with Life patterns will find these algorithms useful in writing efficient Life programs for their computers.

One-Dimensional Life

Jonathan K Millen PhD
661 Main St
Concord MA 01742

The game of Life is known to many computer experimenters for its beautiful, symmetrical two-dimensional displays and for its imaginary population of blinkers, beehives, gliders, and other strange, pseudoliving organisms. Invented by the British mathematician John Conway, the game was described in the "Mathematical Games" section of *Scientific American* in October 1970 and February 1971. A series of articles on how to program it for a home computer also appeared in three of the earliest issues of BYTE ("LIFE Line 1, 2, and 3," BYTE September 1975, page 72; October 1975, page 34; December 1975, page 48). It is an attractive home computer software project, but the program requirements in memory capacity, processor speed and display capability were more than I possessed in my homebrew machine. The programming effort also looked formidable. I developed One-Dimensional Life as a small scale substitute.

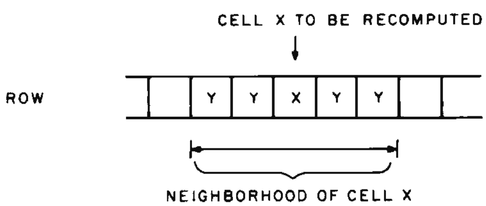


Figure 1: The state of a cell in the next generation is computed from its present state and the states of the four other cells in its neighborhood. The neighborhood of a cell consists of all cells within a distance of two cells from the cell in question.

Conway's Two-Dimensional Life traces successive generations of a pattern of cells in an infinite square array of cells like an uncolored checkerboard. The generation rules determine the state of a cell in the next generation based on its present state and the states of its neighbors, the eight cells touching it.

Each cell has two possible states: off and

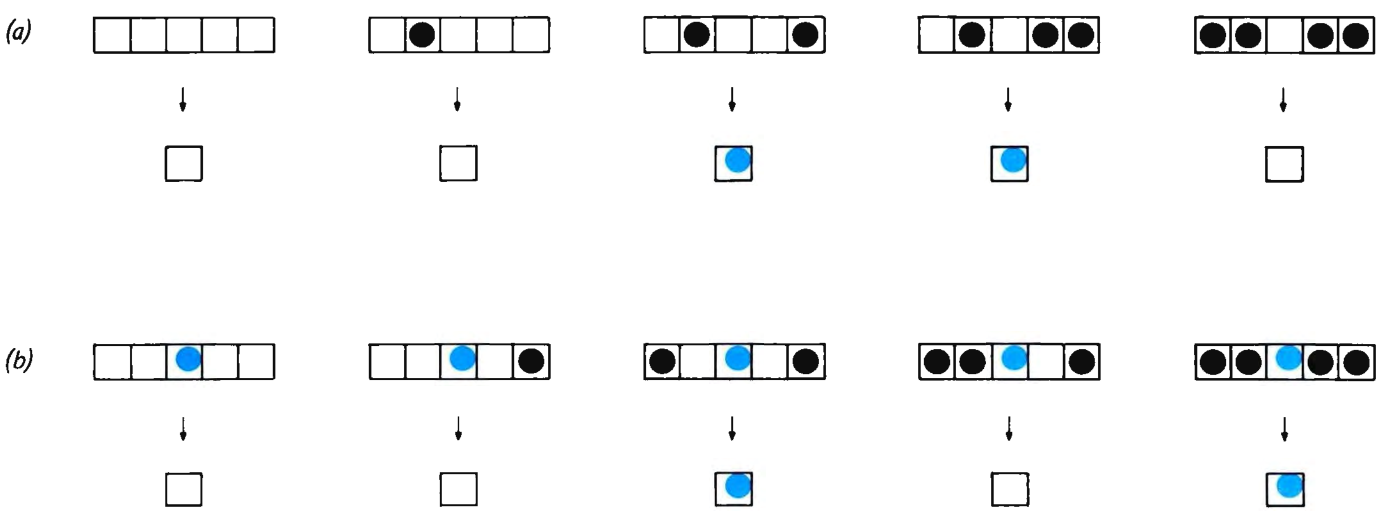


Figure 2: The generation rules, illustrated here for a few representative cases. Each cell is marked with a dot if it is on, and left as an empty square if it is off. The next generation of the middle cell in each neighborhood is shown below it. 2a illustrates the rule that a cell is "born" if and only if it has two or three on neighbors in its neighborhood (in each example, the square being examined is shown in color). 2b illustrates that a cell survives if and only if it has two or four neighboring on cells. Note that a cell dies if it has three on neighbors.

on. A cell is "born" (ie: goes on when previously off) if exactly three of its neighbors are on. A cell survives (ie: stays on) with two *or* three neighbors on. Otherwise it is off in the next generation.

Generation Rules

In a one-dimensional version, patterns have to exist in a single row of cells. Each cell in the row has two cells touching it. I tried all possible generation rules involving a cell and its two neighbors, and I was disappointed in the results with all of them. It finally occurred to me to try a larger neighborhood including not only the adjacent cells but also the two adjacent to them (see figure 1). It still took several tries to come up with generation rules that seemed to yield a game approaching the richness of the two-dimensional game. The rules illustrated in figure 2 met my criteria for interest, which included the existence of oscillating patterns with long periods, patterns with long lifespans that eventually vanish, and gliders. The rules can be summarized as follows: Each cell is viewed with respect to a 5 cell region including itself and two neighboring sights on either side. Cells with two or three neighbors on are born and those with two or four neighbors on survive. The rest are off in the next generation.

The bare rules are rather plain without some biological "facts of life" to dress them up. The following explanation is offered:

> **Rule 1: Birth.** Cells that are off but have either two or three neighbors on, go on.
>
> **Rule 2: Survival.** Cells that are on and have two or four neighbors on stay on. Those with zero or one neighbors on die from loneliness; those with three neighbors on die from overcrowding. What keeps a cell with four neighbors on from dying is not clear. Maybe there is just not enough room to lie down.

Examples

Let us trace the life spans of a few patterns. The simplest oscillating pattern consists of two adjacent cells on. Its next generation has two cells on with two cells off in between, and the third generation regenerates the original pattern. Figure 3 shows three generations of this pattern. Note that the successive generations of a one-dimensional pattern form a two-dimensional pattern.

Another period 2 pattern is the flip flop in figure 4. A line of five adjacent cells on is also periodic, but with period 6. Seven generations of it are shown in figure 5.

A glider is shown in figure 6. It looks the same in every generation, but in each generation it moves one cell to the right. It is easily proved, incidentally, that One-Dimensional Life, unlike Two-Dimensional Life, has no stable patterns that repeat in one generation in the same place.

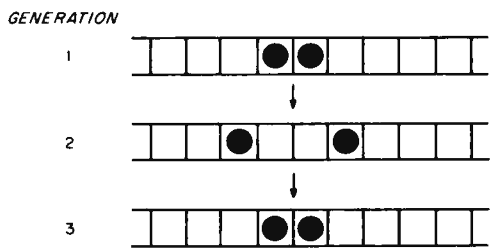


Figure 3: The simplest oscillating pattern, consisting of two adjacent cells on. Three generations are shown. Every second generation recreates the original pattern, so this pattern is said to have period 2. Its alter ego, the pattern with two cells on separated by two cells off, also has period 2. The three generations are separated in order to emphasize that they are separate generations rather than part of a Two-Dimensional Life configuration.

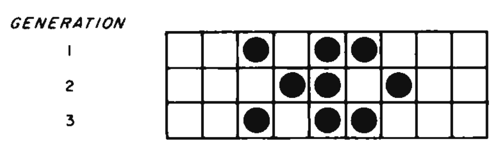


Figure 4: A pattern with period 2 that oscillates between the starting pattern and its mirror image. These kinds of patterns are sometimes called flip flops in conventional Life terminology. (Note that each line is the complete state of the Life universe in one generation.)

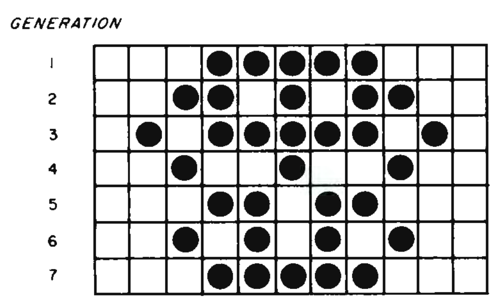


Figure 5: A line of five cells that regenerates itself after six generations. In my experience, this pattern has the longest period of any One-Dimensional Life pattern.

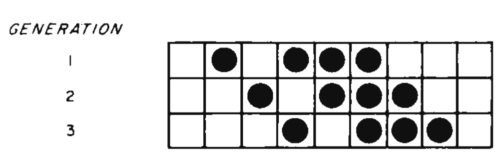


Figure 6: A glider pattern, so called because it regenerates itself in a steadily moving position. This glider has a period of 1.

We have seen a glider of period 1 and two static oscillating patterns of periods 2 and 6. Are there patterns with all possible periods? I generated the life spans of lines with up to 15 cells and found one new oscillating pattern: a line of 12 cells that oscillates with period 4. This suggested that static oscillating patterns could be found for even periods, though I have not yet found any with periods longer than six. Of course if one starts a line of 12 (period 4) and a line of five (period 6) far enough apart to keep them from interacting, the pattern as a whole will not repeat until the twelfth generation, because 12 is the least common multiple of 4 and 6.

What about odd periods? Having found a glider of period 1, I tried a number of similar but longer patterns and discovered the period 3 glider shown in figure 7. Readers may enjoy discovering for themselves another period 3 glider that is one cell longer than the one in figure 7. It is tempting to conjecture that gliders exist with all odd periods. If anyone finds oscillating patterns or gliders of period 7 or greater, I would like to hear about it.

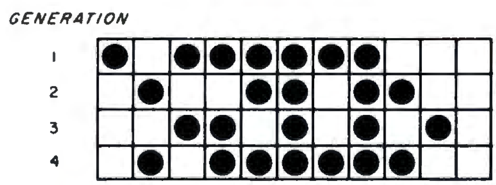


Figure 7: A glider that moves right by one cell every three generations. It has length 8; there is another period 3 glider of length nine. The two intermediate patterns of this glider are, of course, also gliders of period 3.

Pattern Number		Fate
Hexadecimal	Binary	(patterns referenced by hexadecimal)
1	1	Dies after generation 1.
3	11	Oscillates with period 2.
5	101	Dies after generation 2.
7	111	Becomes number 1F after generation 1.
9	1001	Oscillates, second form of number 3.
B	1011	Oscillates with period 2.
D	1101	Oscillates, second form of number B.
F	1111	Becomes number 33 after generation 1.
11	10001	Dies after generation 2.
13	10011	Oscillates with period 2.
15	10101	Oscillates, fourth form of number 1F.
17	10111	Glider with period 1.
19	11001	Oscillates, second form of number 13.
1B	11011	Oscillates, fifth form of number 1F.
1D	11101	Glider with period 1.
1F	11111	Oscillates with period 6.
21	100001	Dies after generation 1.
23	100011	Becomes number 19 after generation 1.
25	100101	Becomes number D after generation 1.
27	100111	Becomes numbers 3F,DB,2B5,FF,37B,A05,201,0.
29	101001	Becomes number B after generation 1.
2B	101011	Becomes numbers 3D,31,13.
2D	101101	Becomes number F after generation 1.
2F	101111	Becomes numbers 23,19.
31	110001	Becomes number 13 after generation 1.
33	110011	Oscillates with period 2.
35	110101	Becomes numbers 2F,23,19.
37	110111	Becomes numbers C7,BF.
.	.	(BF is glider with period 3.)
.	.	
.	.	

Table 1: Vital statistics of the first 28 One-Dimensional Life patterns. The convention used for identifying the patterns is to regard the pattern as a binary number: on cells are represented by 1s, and off cells by 0s. Only the cells between the leftmost and rightmost on cells, inclusive, are considered, and a binary (or hexadecimal) number can then be generated to represent the pattern.

Tabulation

There is an obvious notation for specifying a pattern without drawing a picture. A one-dimensional pattern, being just a sequence of on and off cells, may be regarded as a binary number. By convention, we can consider the pattern proper as just the cells between the leftmost on cell and the rightmost on cell, inclusive. Thus, the number for a pattern will always be odd, except for the "all off" pattern, 0. Pattern numbers can be reported in decimal, octal or hexadecimal to save space. For example, the first generation shown in figure 3 (two adjacent cells) can be represented by binary 11 (or hexadecimal 3). The second generation in figure 3 would then be represented by binary 1001 (or hexadecimal 9). The pattern of binary 11111 (hexadecimal 1F) goes through the following cycles, all noted in hexadecimal: 6B, 17D, 49, 1B, 55 and back to 1F. (Note the 3-digit hexadecimal number 17D, which is needed because the figure is nine cells wide at that point in the cycle.)

This numbering system also provides a handy way of enumerating patterns in a systematic sequence. This gives rise to the idea of constructing a dictionary of patterns in numerical order, listing for each pattern its vital statistics: whether it oscillates, glides, dies or leads to a noninteracting collection of oscillating patterns and gliders. By way of illustration, the first 28 entries in such a table are shown in table 1.

Theoretically there is also the possibility that a pattern may grow in mass (ie: the total number of cells on) without bound, like the glider gun found for Two-Dimensional Life. No such infinitely growing pattern has yet been found for One-Dimensional Life.

Implementation

The discoveries of the period 4 pattern and the large period 3 gliders were made on my Turing machine computer. Implementing the One-Dimensional Life rules was easy and would probably have been easier if

MICROFILE

A New Datafile Management Program With A Comprehensive Concept In Software Presentation Offered By:

PRS

MICROFILE MAKES BREAKTHROUGH IN DOCUMENTATION

PRS "Microfile" comes with a complete and instructive handbook explaining every procedure step by step. This unique documentation is written in clear and easy-to-understand English and avoids esoteric technical language.

But PRS "Microfile" documentation is more than clear and precise instructions in a graphically appealing manual. You learn at the same time. As you and your computer work together, the text helps you to create interesting applications.

PRS solves one of the major concerns of beginners. You will find a special chapter on how to interface newly acquired software with your specific configuration. PRS gives clear tutorial instructions on the whole matter of I/O patches. In addition, you will always find a complete source listing of the I/O section of our programs. PRS makes every effort to facilitate the implementation of its software within your particular system.

Indeed, PRS "Microfile" documentation offers you a true fluency in computer-user dialogue. This in turn allows you to arrive at an efficient and effective use of your PRS "Microfile" program and microcomputer.

PRS MICROFILE HAS A POWERFUL CODE

Powerful code means fast and accurate answers. PRS permeates the design of its unique code with "human engineering." And you'll appreciate the delicate balance between machine-human dialogue and resident program-space memory.

PRS MICROFILE COMES IN BEAUTIFUL CASSETTE-FOLDER

Your PRS "Microfile" cassette and manual is protectively nestled in an efficient and durable gold-stamped folder. Wherever you file it, in your room or office, the decor will be enhanced by its elegance.

WITH PRS MICROFILE, A NEW DATAFILE MANAGEMENT PROGRAM YOU CAN:

- Manage any list of items
- Edit information already on file
- Sort items alphanumerically upwards and downwards
- Search by primary and/or secondary keys (labels)
- Total sums of columns
- Justify right or left margins of each column
- Save and retrieve data with mass storage media

UNLIMITED APPLICATIONS

For example, PRS "Microfile" manages:

- Home inventory
- Listing of musical tapes
- Calendar of activities
- Vacation itineraries
- Checks by categories
- Home-upkeep tickler files
- Family historical data
- Medical Records

PRS MICROFILE SUPPORTS YOUR MAJOR BRANDS

- Cuts cassette version: runs as is with SOLOS/CUTER monitor easily adaptable to the SORCERER from EXIDY.
- TOL cassette relocatable Z-80 version: runs with Zapple monitor.
- TARBELL cassette version: interfaced to the APPLE monitor (8080A version of ZAPPLE) but easily adaptable to your own monitor (see above paragraph on our documentation.)
- TRS-80 version: runs on level II with 16 K minimum.
- ICOM 8" disk CP/M version.
- MICROPOLIS 5" disk CP/M version.

VISIT YOUR NEAREST DEALER TODAY TO OBTAIN YOUR PRS MICROFILE FOR ONLY $49.95 FOR CASSETTES AND $64.95 FOR FLOPPY DISKS.

Let him demonstrate the excellence of PRS "Microfile", a new DATAFILE MANAGEMENT SYSTEM. Or use the coupon below to obtain the name of a dealer in your area.

New Product Releases From

PRS THE PROGRAM OF THE MONTH CORPORATION
257 Central Park West, New York, N.Y. 10024

Gentlemen:

☐ Please place my name on your priority mailing list to receive your descriptive advance releases of all new programs developed by **PRS** at regular intervals.
☐ Please send me a list of dealers in my area.

Name ______________________

Title ______________________

Corporation ______________________

Address ______________________

City ____________ State ________ Zip ________

Telephone ______________________ Reference 12B8

Circle 309 on inquiry card.

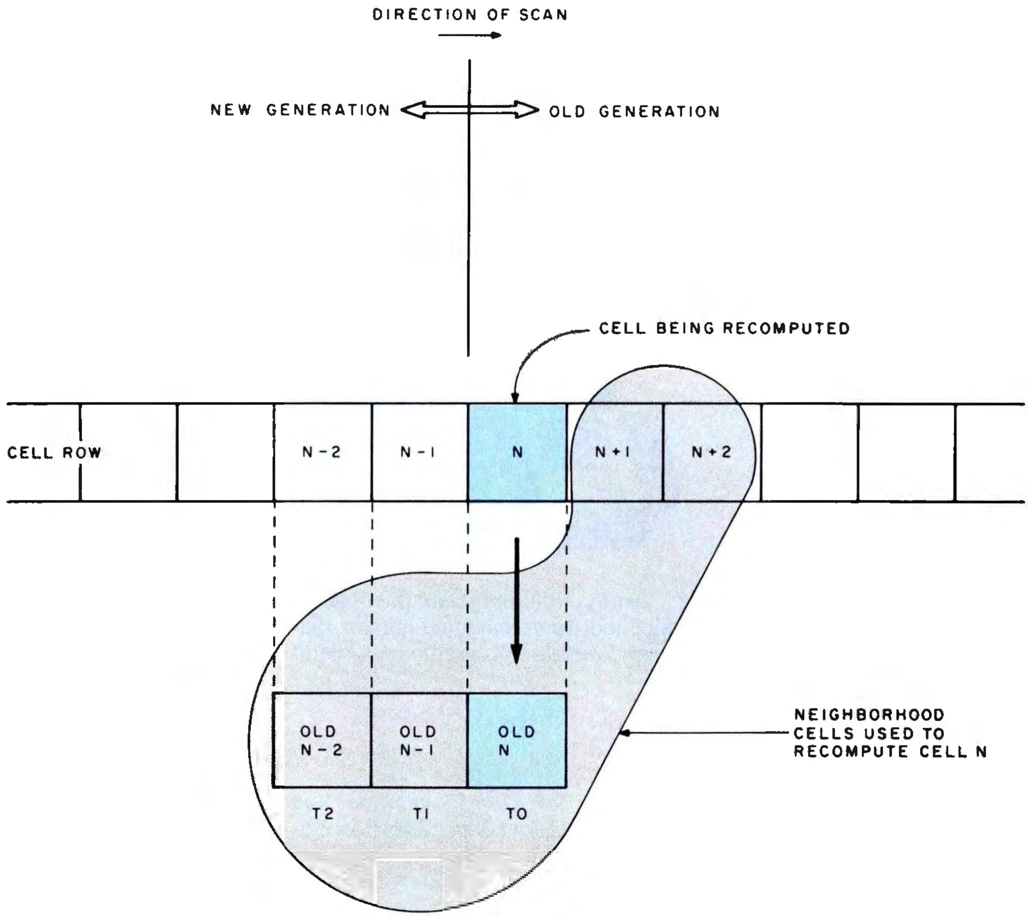
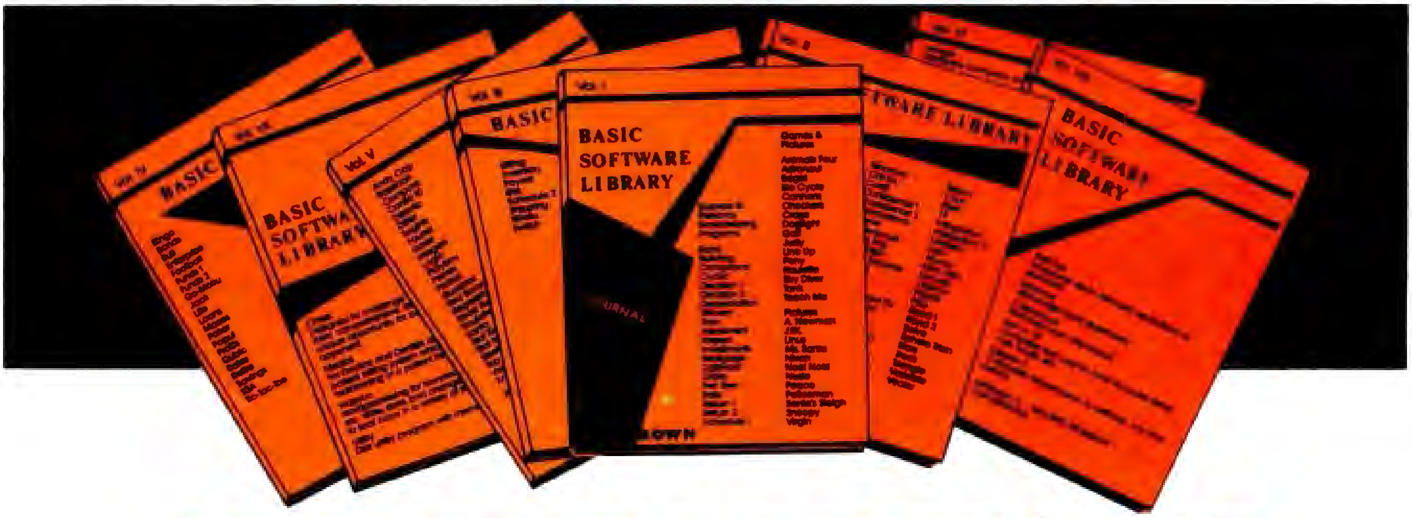


Figure 8: Calculating the next generation in One-Dimensional Life. The algorithm uses a row of cells and three temporary cell variables. Each cell has one of two values: 0 for off and 1 for on. Each cell is a bit, a byte, or a word, whichever is most convenient in the programming language to compute the algorithm. The states of the cells are recomputed in a scan from left to right. Cells behind the front of the scan have already been recomputed; those ahead of it still have their old states. As the algorithm scans from left to right, it must temporarily remember the old states of each cell long enough to compute the new values of its two neighbors to the right. (Similar temporary value storage problems arise in conventional Life programs.)

I had had a more conventional computer to work with, instead of Turing machine (see "A Universal Turing Machine," December 1976 BYTE, page 114). The algorithm for producing new generations according to the rules is illustrated in figure 8, visualizing the memory requirements, and figure 9, a flowchart. Note that, as in Two-Dimensional Life, there is one complication to bear in mind when computing the next generation: When you change the state of a cell, you must remember the old state long enough to use it in computing the next state of its neighbors. If the program scans the row from left to right, changing cells as it goes, it needs a temporary memory of three cells. When the front of the scan is at cell N, the program is able to recompute cell N, after saving it, using its memory of the prior states of cells N - 1 and N - 2, together with the present states of cells N, N + 1 and N + 2. The old state of N - 2 may then be forgotten and the scan moved right one cell.

I will spare you the details of how this algorithm can be accomplished in Turing machine language. A more universal problem is how to get the patterns displayed. My only output device, at first, was a single LED (light emitting diode) that could be stepped through memory, to display it one bit at a time. To improve on this I built a visible shift register, a cascade of two 8 bit shift registers with an LED on each output, giving me a movable 16 bit window on memory. I was considering extending the window to 32 bits when I was lucky enough to get a long term loan of a SwTPC CT-1024 video display from a friend who had no present need of it.

There is a coincidental resemblance between a Turing machine and a video display: both normally change memory addresses by ± 1. It turned out to be easy and natural to patch in the CT-1024 memory in place of the Turing machine "tape" memory. Turing machine computations were then directly visible as they progressed on the video screen. One more refinement was all that was needed to display successive generations of a pattern below one another as in the


BASIC SOFTWARE LIBRARY

Figure 9: Flowchart for the One-Dimensional Life algorithm.

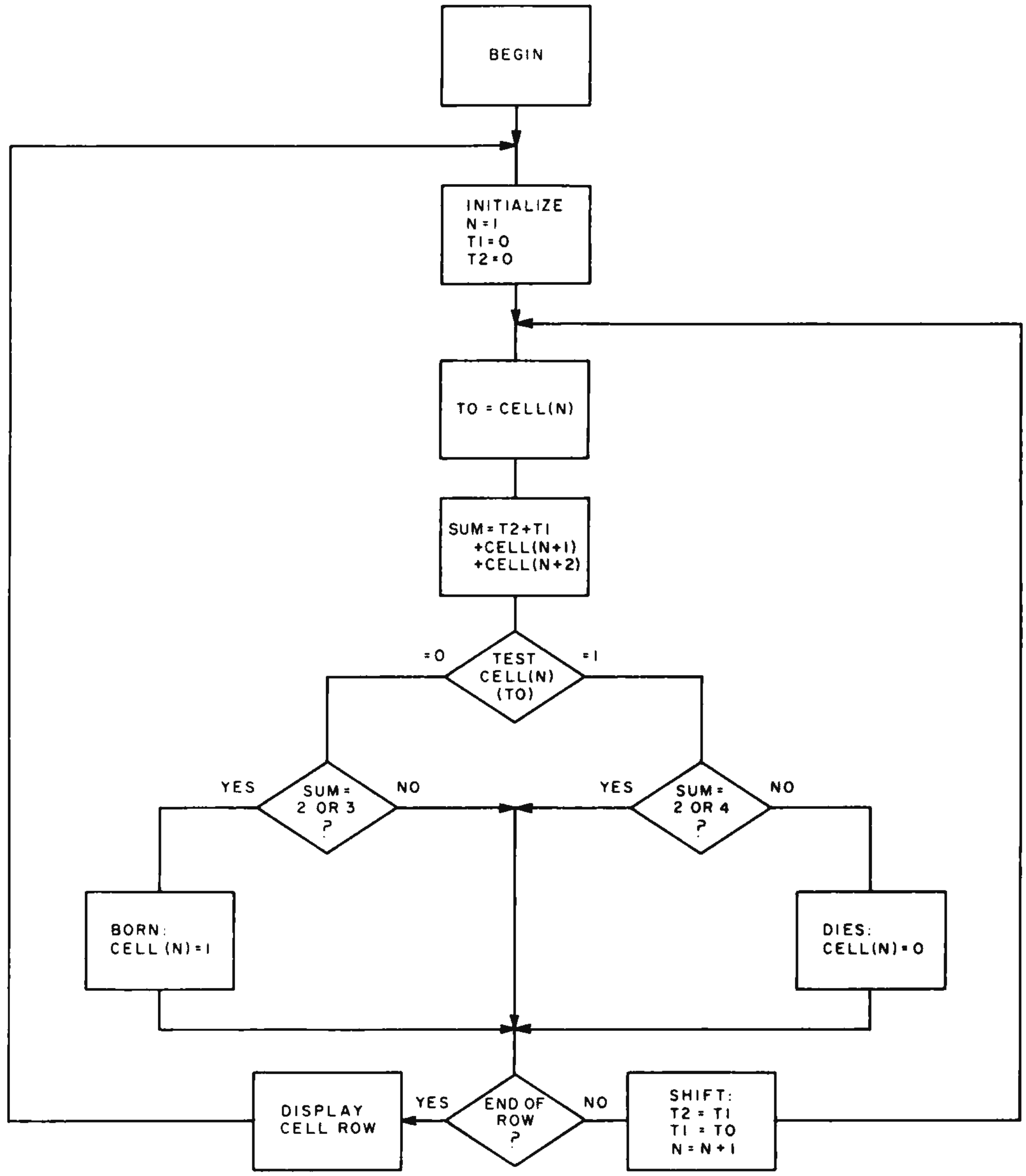
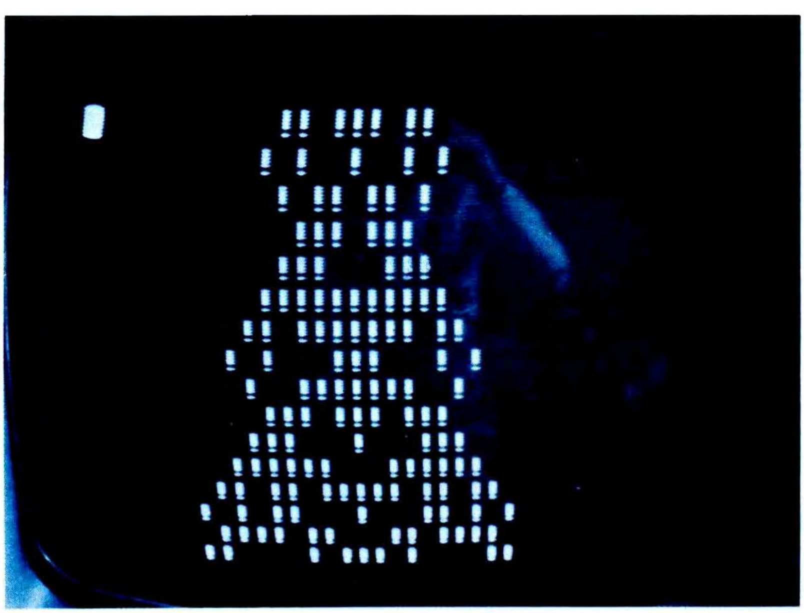


Photo 1: One-Dimensional Life display. On the author's system, 1s are represented by exclamation points and 0s by blanks. Each line represents one generation of One-Dimensional Life.



figures: up and down cursor control. Outputs were created by decoding the last three instruction address bits, effectively yielding eight 1 bit output ports that were strobed every time an instruction with the appropriate address was executed. Two of these outputs became cursor control outputs.

After all this hardware activity and some program modifications, the result was the kind of display shown in photo 1. The screen has the first 16 generations of a line of seven cells. This pattern settles down after 40 generations to a collection of noninteracting oscillating patterns, but before that happens it produces one of the most intriguing displays of its kind, one that I would never have seen without the help of a home computer. Of course, readers can verify these discoveries with *any* home computer and share in some of the excitement of exploration which I found.■

Software for the Percom LFD-400

Disk Operating and File Management Systems

INDEX™: The most advanced Disk Operating and File Management System available for the 6800. INterrupt Driven EXecutive operating system features file-and-device-independent, queue-buffered character stream I/O. Linked-File disk architecture, with automatic file creation and allocation for ASCII and binary files, supports sequential and semi-random access disk files. Multilevel file name directory includes name, extension, version, protection, and date. Requires 8K RAM at $AOOO. Diskette includes numerous utilities. $99.95

MINIDOS-PLUSX: An easy to use DOS for the small computing system. Supports up to 31 named files. Available on ROM or diskette complete with source listing. $39.95

Basic Interpreters and Compilers

SUPER BASIC: a 10K extended disk BASIC interpreter for the 6800. Faster than SWTP BASIC, 9-digit accuracy, program CHAINing, BASE O subscripting, improved error reporting, disk data files, and print column alignment. Program and data files may be prepared using one of the Text Editors described below $49.95

BASIC BANDAID™: Turn SWTP 8K BASIC into a respectable random access data file disk BASIC. Includes many speed improvements and program CHAINing. When ordering, specify version (2.0, 2.2, or 2.3). Complete with listing. $17.95

STRUBAL™: A STRUctured BAsic Language Compiler for the serious professional programmer. Includes elements of BASIC, PL/M, and assembly language. Features 10-digit floating point, strings, scientific functions, and 2-dimensional arrays. Requires 16K RAM memory and LINKING LOADER (see below). Complete with RUN-TIME and FLOATING POINT packages $99.95

Text Editors

EDIT68: Hemenway Associates powerful disk-based text editor. May be used to create programs and data files. In addition to SEARCH, CHANGE, DELETE, and MOVE functions, EDIT68 supports MACROs which perform complex repetitive editing functions efficiently. Runs in only 6K RAM. "Spooling" permits text files much larger than the available RAM memory to be created and edited. $29.95

TOUCHUP™: If you already have the TSC Text Editor, TOUCHUP will convert it into a disk-based text editor. ROLL function permits text files much larger than the available RAM memory to be created and edited. TOUCHUP is supplied on diskette complete with source listing. $17.95

ASSEMBLERS

PERCOM 6800 SYMBOLIC ASSEMBLER: Requires only 8K of RAM memory to assemble programs with 200 labels. More convenient to use than most assemblers; assembly options may be specified at time of assembly. $29.95

Complete listing of above assembler $29.95

MACRO-RELOCATING ASSEMBLER: This Hemenway Associates assembler is for the programming professional. Generates relocatable and linkable object code. MACRO facility permits nested macro calls. Permits conditional assembly. Requires 16K RAM $49.95

LINKING LOADER for STRUBAL and the above assembler $19.95

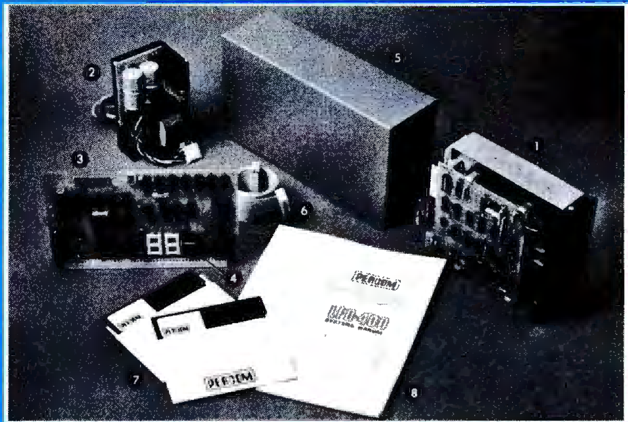
Business Applications

GENERAL LEDGER SYSTEM $199.95
FULL-FUNCTION MAILING LIST .... $ 99.95

Write or call for our complete software catalog.

PERCOM DATA COMPANY, INC.
Dept BP • 318 Barnes • Garland TX 75042
(214) 272-3421

For Your SWTP 6800 Computer...



PERCOM's™ Assembled and Tested FLOPPY DISK SYSTEM

the LFD-400

only 599^{95} shipping paid

The LFD-400 is ready to plug in and run the moment you receive it. Nothing else to buy! Not even extra memory!

YOU GET:

1. The popular Shugart SA 400 minifloppy™ drive. Drive alignment is double checked by PerCom before shipment.
2. The drive power supply—fully assembled and tested.
3. LFD-400 Controller/Interface—plugs into the SS-50 bus • accommodates three 2708 EPROMs • fully assembled and tested.
4. MINIDOS™—the remarkable LFD-400 disk operating system on a 2708 EPROM • plugs into the LFD-400 Controller card • no extra memory required • no "booting" needed.
5. Attractive metal enclosure.
6. Interconnecting cable—fully assembled and tested.
7. Two diskettes—one blank, the other containing numerous software routines including patches for SWTP 8K BASIC and the TSC Editor/Assembler.
8. 70-page instruction manual—includes operating instructions, schematics, service procedures, and the complete listing of MINIDOS™.
9. Technical Memo updates—helpful hints which supplement the manual instructions.
10. 90-day limited warranty.

Minifloppy is a trademark of Shugart Associates.
MINIDOS is a trademark of PERCOM Data Company, Inc.

The LFD-400 is readily expanded to either two or three drives. Write for details. Send for our free brochure for more information about the LFD-400 Floppy Disk System and LFD-400 software.

To save you money, the LFD-400 Floppy Disk System is available only from PerCom. Because of the special pricing, group and dealer discounts are not available.

MC and VISA welcome. COD orders require 30% deposit plus 5% handling charge. Allow three extra weeks if payment is by personal check. The LFD-400 Floppy Disk System is available immediately. Allow three weeks for testing and transportation. Texas residents add 5% sales tax.

PERCOM™

PERCOM DATA COMPANY, INC.
Dept B 318 BARNES • GARLAND, TX. 75042
(214) 272-3421
PERCOM™ *'peripherals for personal computing'*

Programming Quickies

William Englander
1966 Titus St
San Diego CA 92110

Life

The game of Life was developed by John Horton Conway and was introduced in the "Mathematical Games" section of the October 1970 *Scientific American* magazine. Life is played on a grid of squares (in this case a 22 by 22 matrix). A given square is either occupied or empty. The program user specifies which squares are occupied initially.

The game of Life program produces new generations of the matrix by applying life's laws for birth, survival and death to the present generation. These laws are:

Birth: An unoccupied square becomes occupied if in the preceding generation exactly three of the eight neighboring squares were occupied (squares that touch horizontally, vertically or diagonally are said to be neighboring squares).

Survival: An occupied square remains occupied if in the preceding generation two or three neighboring squares were occupied.

Death: An occupied square becomes unoccupied if in the preceding generation fewer than two or more than three neighboring squares were occupied.

Text continued on page 82

```
BASIC-E COMPILER  VER 1.4
      DIM A(22,22),B(22,22)
      INPUT 'ENTER INITIAL NUMBER OF ITERATIONS';L
      PRINT 'ENTER INITIAL COORDINATES; 0,0 TO END'

 10   INPUT X,Y
      IF X+Y=0 THEN GOTO 20
      IF (X<1)OR(X>20)OR(Y<1)OR(X>20) THEN PRINT 'ERROR RE-ENTER': GOTO 10
      X=X+1: Y=Y+1
      A(X,Y)=A(X,Y)+10
      FOR XW=X-1 TO X+1: FOR YW=Y-1 TO Y+1
          A(XW,YW)=A(XW,YW)+1
          NEXT YW: NEXT XW
      GOTO 10

 20   L$='+------------------------------------------------+'
      INPUT 'ENTER PAPER SIZE (IN LINES/PAGE), SET UP PAPER & HIT ENTER';P
      FOR I=1 TO L STEP 2

          PRINT L$
          FOR Y=1 TO 22: PRINT '|';: FOR X=1 TO 22
              IF A(X,Y)<10 THEN PRINT '  '; ELSE PRINT '[]';
              B(X,Y)=0
              NEXT X: PRINT '|': NEXT Y
          PRINT L$
          FOR J=25 TO P: PRINT: NEXT J

          FOR X=2 TO 21: FOR Y=2 TO 21
              AW=A(X,Y)
              IF (AW<>3)AND(AW<>13)AND(AW<>14) THEN GOTO 30
              B(X,Y)=B(X,Y)+10
              FOR XW=X-1 TO X+1: FOR YW=Y-1 TO Y+1
                  B(XW,YW)=B(XW,YW)+1
                  NEXT YW: NEXT XW
 30           NEXT Y: NEXT X

          PRINT L$
          FOR Y=1 TO 22: PRINT '|';: FOR X=1 TO 22
              IF B(X,Y)<10 THEN PRINT '  '; ELSE PRINT '[]';
              A(X,Y)=0
              NEXT X: PRINT '|': NEXT Y
          PRINT L$
          FOR J=25 TO P: PRINT: NEXT J

          FOR X=2 TO 21: FOR Y=2 TO 21
              BW=B(X,Y)
              IF (BW<>3)AND(BW<>13)AND(BW<>14) THEN GOTO 40
              A(X,Y)=A(X,Y)+10
              FOR XW=X-1 TO X+1: FOR YW=Y-1 TO Y+1
                  A(XW,YW)=A(XW,YW)+1
                  NEXT YW: NEXT XW
 40       NEXT Y: NEXT X
      NEXT I

      INPUT 'ENTER NUMBER OF ADDITIONAL ITERATIONS';L
      IF L>0 THEN GOTO 20
      STOP
      END
 0 ERRORS DETECTED
```

Listing 1: BASIC E program and sample run of the game of Life. A sequence of eight states of Life demonstrates operation of the program.

MICROPRO INTERNATIONAL CORPORATION

MICROPRO

MicroPro International Corporation™

professional quality you can count on!

proudly announces

SUPER-SORT©

The ultimate in high performance sort/merge

pecifications

-]High Performance Tournament/Heapsort Algorithm
-]Multiple Input Files Read in Parallel for True Merge Operation. Up to 32 Sort and 32 Merge Input Files Handled.
-]Dynamically Invokable User Exit Routines
-]Record Selection via SELECT/EXCLUDE Statements
-]Handles Fixed *and* Variable Length Records
-]Handles Fixed *and* Variable Length Fields
-]Handles up to 32 Sort Keys with Intermixed Sequence Indicators, Data Types and Alternate Collating Sequences
-]Handles Multi-Volume Diskette Files
-]Compatible with CP/M* and any Derivative Including ADOS, IMDOS, CDOS, etc.
-]8080/8085/Z-80 Compatible
-]Specialized Optimizations for Floppy Disk Environment
-]Keyword Command Input for Easy Operator Entry
-]Benchmarked at Over 560 Records per Minute!
-]Invokable as a Subroutine from FORTRAN, COBOL and Assembler
-]Furnished in Relocatable and Executable Form for Easy Load-Address Definition
-]Optional TAGSORT Operation
-]Data Types Include ASCII, EBCDIC, Binary, BCD (COBOL Packed Decimal), etc.
-]Supports CP/M-compatible Diskette Files under BASIC, FORTRAN, COBOL and Assembler.

'rice $250** includes manual and single densitydiskette.
Aanual only $15 refundable with purchase.

WORD-MASTER©

The last word in text editing

Specifications

- ☐Compatible with any "Dumb" CRT Possessing Addressable Cursor and Backspace (Includes Hazeltine - All Models, SOROC, Lear Siegler, IMSAI-VIOC, ADDS Regent, Behive, etc.)
- ☐Bi-Directional Word Tab, Line Tab, Screen Tab
- ☐Bi-Directional Word Delete, Line Delete and Character Delete
- ☐Quad-Directional Cursor Movements
- ☐Mid-Line Insert and Delete
- ☐Automatic RAM/Diskette Buffering With No User Intervention
- ☐Nested Command Looping with Conditional Execution
- ☐Global String Search, Global String Replace
- ☐Scratch-Pad Buffer for Text Movement, Global Replication, and String Command Storage
- ☐Multiple Input File Merging with User-Controlled Insertions for Easy Document Assembly
- ☐Multiple Output File Control by Section Under User Control
- ☐Compatible with CP/M and Most of its Derivatives Including CDOS, IMDOS, etc.
- ☐Use TEX* for Print Formatting Functions

Price $150** includes manual and single density diskette.
Manual only $9.00 refundable with purchase.

*CP/M and TEX are Trademarks of Digital Research.

** Prices and Specifications subject to change without notice.

ealer Inquiries Invited: Call (Northern California) (707) 544-2865, (415) 398-7062, (209) 445-0511, (408) 279-8980, (916) 485-619 (Southern California) (213) 224-1619, (714) 634-2908. Outside California Call Collect (707) 544-2865. Principal offices cated at 5810 Commerce Blvd., Rohnert Park, CA 94928

Circle 224 on inquiry card

About the Author

William Englander is a self-employed computer programmer as well as an instructor at San Diego State University and National University.

Listing 1, continued:

```
  RUN LIFE
BASIC-E INTERPRETER - VER 1.3


ENTER INITIAL NUMBER OF ITERATIONS? 8
ENTER INITIAL COORDINATES; 0,0 TO END
? 8,10
? 9,10
? 10,10
? 11,10
? 12,10
? 0,0
ENTER PAPER SIZE (IN LINES/PAGE), SET UP PAPER & HIT ENTER? 33

+-------------------------------------------+
|                                           |
|                                           |
|            [][][][][]                     |
|                                           |
|                                           |
+-------------------------------------------+

+-------------------------------------------+
|                                           |
|                                           |
|              [][][]                       |
|              [][][]                       |
|              [][][]                       |
|                                           |
|                                           |
+-------------------------------------------+

+-------------------------------------------+
|                                           |
|                [ ]                        |
|              [ ]   [ ]                    |
|            [ ]       [ ]                  |
|              [ ]   [ ]                    |
|                [ ]                        |
|                                           |
+-------------------------------------------+
```

Listing 1, continued:

```
+--------------------------------------------+
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                      []                    :
:                    [][][]                  :
:                  [][]  [][]                :
:                    [][][]                  :
:                      []                    :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
+--------------------------------------------+


+--------------------------------------------+
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                  [][][]                    :
:                [][]    [][]                :
:                []      []                  :
:                []      []                  :
:                  [][][]                    :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
+--------------------------------------------+


+--------------------------------------------+
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                    []                      :
:                  [][][]                    :
:                []  []  []                  :
:              [][][]  [][][]                :
:                []  []  []                  :
:                  [][][]                    :
:                    []                      :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
:                                            :
+--------------------------------------------+
```

Listing 1, continued:

```
+----------------------------------------------+
:                                              :
:                                              :
:                                              :
:                   [][][]                     :
:                                              :
:               []          []                 :
:               []          []                 :
:               []          []                 :
:                                              :
:                   [][][]                     :
:                                              :
:                                              :
:                                              :
+----------------------------------------------+

+----------------------------------------------+
:                                              :
:                                              :
:                                              :
:                      []                      :
:                      []                      :
:                      []                      :
:                                              :
:              [][][]      [][][]              :
:                                              :
:                      []                      :
:                      []                      :
:                      []                      :
:                                              :
:                                              :
+----------------------------------------------+
```

Text continued from page 76

The Life program in listing 1 was written in BASIC E and run on an IMSAI 8080. Since it is necessary to reference the present generation's matrix while developing the next generation's matrix, two arrays, A and B, are used alternately. When an array element represents an occupied square, it is given a value of 10. 1 is added to it for each occupied neighboring square (including itself for convenience). Consequently a square in the next generation becomes occupied if its corresponding element in the present generation array is equal to 3, 13 or 14 (an empty square with three neighbors or an occupied square with two or three neighbors).

Statements 1 through 13 establish the number of generations to be printed and the initial occupied squares (in the A array). Statements 19 through 25 print the contents of the A array and zero the B array. Statements 27 through 34 generate the B array from the A array. Statements 36 through 42 print the B array and set the A array to zero. Statements 44 through 52 generate the A array from the B array and then loop back to produce the next two generations.■

Thinking of adding mass memory to your S-100 bus?

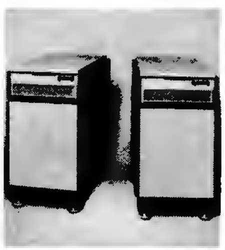
Think twice!

You say you're ready for greater storage? Well now, with average access times of 28 milliseconds, large, on-line direct access cartridge disk files have become a reality in microcomputing. And, ALPHA MICRO offers you not one, but TWO ways to vastly increase your system capacity...and do it without overtaxing your budget.

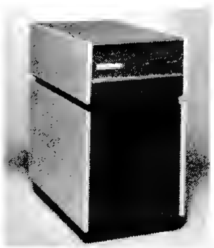
First, there's the ALPHA MICRO AM-500™ Hard Disk Subsystem. It uses the popular CDC 9427H (Hawk) cartridge disk drive with a total of 10 megabyte capacity (5 fixed, 5 removable). The AM-500 comes complete with interface formatter/controller, cabling, and disk drive. You can expand your mass memory to meet your requirements, in 10 megabyte increments, up to 40 megabytes.

Thinking of even more mass memory? Check out the ALPHA MICRO AM-400™ Hard Disk Subsystem. It features the CALCOMP TRIDENT Series Hard Disk Drive in a choice of models with 25, 50, 80, 200, or 300 megabyte capacity. And, you can daisy-chain up to four units, on-line, in any mix. How's that for capacity?

So, if you're thinking of adding mass memory to your S-100 bus, be sure to check out the ALPHA MICRO AM-400 or AM -500 Hard Disk Subsystem at your nearest ALPHA MICRO Dealer. And while you're there, ask him to show you the rest of the ALPHA MICRO hardware and software lines.



AM 400™



AM 500™

alpha micro

17881 Sky Park North
Irvine, California 92714
(714) 957-1404

Circle 4 on inquiry card.

Chess 4.7 versus David Levy

The Computer Beats a Chess Master

J R Douglas
25 Cannon Dr
Elk River MN 55330

After 29 years, computer chess finally achieved a victory in human competition at the master class tournament level. During the fourth game of a match held at the Canadian National Exhibition from August 26 to September 4 1978, International Master David Levy resigned to Chess 4.7/ CYBER 176 after 56 moves, although he did win the tournament, 3½ to 1½.

David Levy was three years old in 1949, when the American mathematician and computer science pioneer Claude Shannon produced the first paper describing the methodology for producing chess playing computer programs. Not until 1956 did any machine win a game against a human opponent: MANIAC, a system developed at the Los Alamos Scientific Laboratory, won a greatly simplified chess game against a novice player in 23 moves.

12 years later, Levy, expert rated and Scottish National Champion, attended the Fourth Annual Machine Intelligence Workshop. There he took exception to the views of John McCarthy of Stanford University and Prof Donald Michie of Edinburgh University, who agreed that within ten years a computer system would be World Champion of chess. Levy countered that not only would computers fall short of that goal, but they would be unable to defeat *him* in a tournament style match within that 10 year period. Neither side was able to shake the other's convictions and, as a result, Levy wagered £1250 sterling that he could defend against the computer advances.

The machine intelligence community had expected Levy to be defeated by a large network of computers participating in the game, until 1970, when a Northwestern University program called Chess 3.0, written by Larry Atkin, Keith Gorlen and David Slate, clearly emerged as the leading effort in the first US Computer Chess Championship. David Levy was then 24.

The original feeling of confidence Levy held must have been somewhat shaken as the years 1973 and 1974 saw Chess 4.0 achieve a United States Chess Federation rating higher than that of the average US tournament chess player. [**Note:** *the version number of the program increases along with its skill.*] Then, in 1976 and 1977, when Chess 4.5 and 4.6 won the class B championship at the Paul Masson Open Chess Tournament and won outright at the Minnesota Open, Levy conceded that he had begun to think that his match with Chess 4.7, "would not be a formality but could be just a bit of work."

The latter part of 1977 and early 1978 saw a series of 2 game matches between Levy and Chess 4.6, the Duchess program from Duke, Greenblatt's MIT program, and Kaissa from the USSR. Levy handily defeated all the programs in the first game.

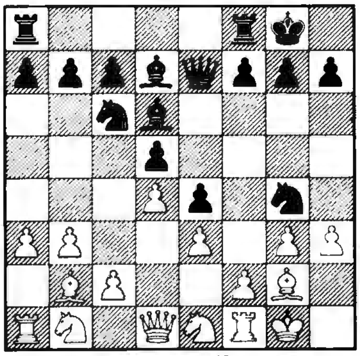
Chess 4.7, running on a Control Data Corp (CDC) CYBER 176, had compiled a rating of 2030 after 31 tournament games and a speed chess performance rating of 2450, when the last challenge was given. The issue was to be resolved on the tenth anniversary of the original wager, with play to begin on Saturday, August 26.

Getting a computer to a chess match, which was the duty of this author and Dr Dave Cahlander, is a considerably more difficult task than getting a human to a match. Crossing the Canadian border with microprocessor controlled chessboards, and setting up and testing telephone lines and modems between Toronto and the CYBER 176 in Arden Hills MN consumed most of a week.

The glass box in which the match was held, standing beside three bowling lanes and a fencing exhibition, faced a large demonstration chessboard and seats for onlookers. A square of chess tables used in simultaneous play filled the rest of the room. Opposite the glass box was the stand

Photo 1: International Master David N L Levy ponders his move while sitting in the glass enclosed booth at the Canadian National Exhibition. A crucial position in one of the match games appears on the electronic chessboard, which is connected by telephone lines to a Control Data Corp CYBER 176 computer.

GAME I



POSITION AFTER 12. P-KR3

Figure 1: Position occurring in round 1 after White's 12th move. The player of Black next unleashes an attack which wins material and disrupts White's Kingside.

1.	P-KN3	P-Q4	33.	NxR	RxN
2.	B-N2	P-K4	34.	B-N4	R-KB6
3.	P-Q3	N-KB3	35.	R-Q8	P-KR3
4.	N-KB3	N-B3	36.	RxP	RxP
5.	O-O	B-Q2	37.	R-Q8	R-KB6
6.	P-N3	B-QB4	38.	R-R8	P-KN4
7.	B-N2	Q-K2	39.	P-Q5	P-KR4
8.	P-QR3	P-K5	40.	P-Q6	K-N2
9.	N-K1	O-O	41.	RxP	R-B2
10.	P-Q4	B-Q3	42.	R-R5	K-B3
11.	P-K3	N-KN5	43.	B-B3 check	K-N3
12.	P-R3	NxP/6	44.	R-K5	R-B6
13.	PxN	Q-N4	45.	B-N4	R-B5
14.	P-KN4	QxP/6 check	46.	R-K7	R-B2
15.	R-B2	B-N6	47.	RxP/4	R-Q2
16.	Q-K2	QxR check	48.	R-K7	P-R5
17.	QxQ	BxQ check	49.	K-N2	P-N5
18.	KxB	P-B4	50.	K-R2	P-N3
19.	PxP	N-K2	51.	K-N2	R-Q1
20.	P-B4	RxP check	52.	P-R4	N-Q2
21.	K-N1	P-B3	53.	P-R5	N-B3
22.	N-QB3	R-R4	54.	PxP	N-Q4
23.	K-R2	R-KB1	55.	P-N7	NxR
24.	N-Q1	N-N3	56.	PxN	R-KR1
25.	R-B1	BxP	57.	B-Q6	K-B3
26.	BxB	R-B8	58.	P-N8=Q	RxQ
27.	N-N2	R-B6	59.	BxR	KxP
28.	PxP	R/4XB check	60.	B-B4	K-B3
29.	K-N1	PxP	61.	B-Q2	K-N3
30.	R-B8 check	N-B1	62.	B-K1	K-N4
31.	B-B3	R-Q6	63.	B-B2	K-R4
32.	N/1-K3	R/RxN		Game agreed drawn.	

Table 1: The score (record of moves) of game 1 of the match. The reader is asked to examine this game, and to form an opinion concerning which player had which color of pieces.

About the Author

J R Douglas has 16 years of experience as a microprogrammer, and maintains an interest in artificial intelligence. His hobbies include photography and amateur radio (callsign KA0ACN).

of Josef Smolij, local speed chess king and guru of the all-night, outdoor Yonge Street Chess Association. Josef, we were to learn, would play a large part in the first win ever for a chess machine at the master level.

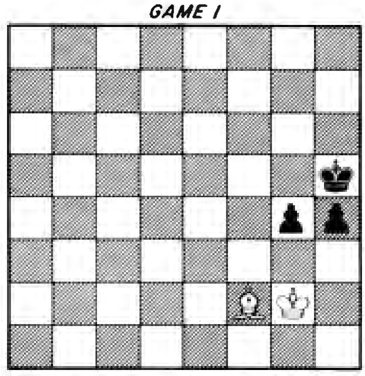
The relationship between the opponents in the Levy match is difficult to describe. The two Davids, Levy and Slate, and the CDC folks stayed in the same hotel and ate meals, travelled and generally spent the entire time together as friends. Levy even considered the machine to be sort of a friendly foe. Each night the entire group found itself on the sidewalks of Yonge Street playing chess on overturned milk cartons with Joe Smolij until the small hours of the morning. Joe demonstrated his "Smash-Crash" Gambit (also known as the Greco Counter Gambit for those who have not yet met Josef) for 50 cents a lesson.

Levy's plan for the match was not difficult to anticipate, since he had demonstrated that, while tactical positions favored the computer, strategic positions favored him. He had used close, quiet games to defeat the computers in each defense of the wager, playing a strategic game until a weakness developed in the computer's position, then winning against that weakness.

The game score of round 1 is presented in the form of a Turing experiment. For those not familiar with him, Alan Turing proposed a method for determining whether a machine should be called "intelligent." In this test, a human, linked via teletypewriter with a machine, is told that he is communicating with either a machine or another human. If he is unable to determine with which of them he is communicating, the machine can be termed "intelligent." The question: was Levy White or Black in game 1? Consult table 1 and form an opinion. The answer appears in the text box on page 90.

The first game was a draw. This created a great deal of speculation, as most of the

GAME I



POSITION AFTER 63. ...K-R4

Figure 2: The final position reached in game 1. The participants agreed to a draw.

ComputerLand®

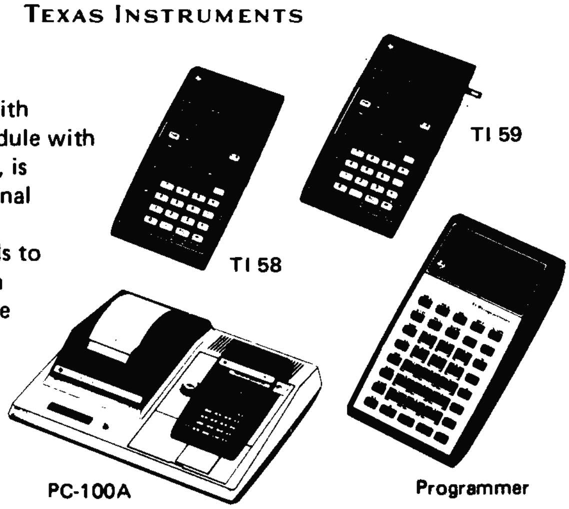
BUY NOW PAY LATER

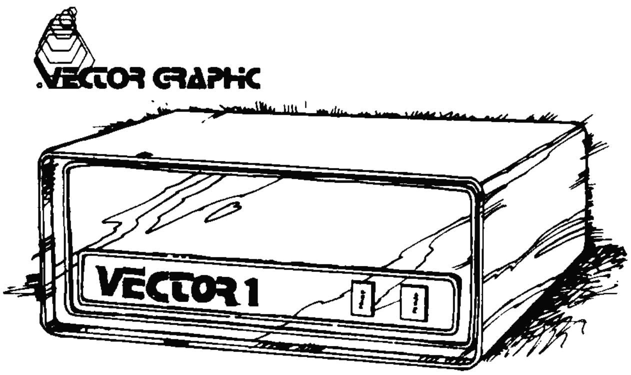
Innovations in Calculator Technology

Advancements from Texas Instruments are available in many forms at ComputerLand. Like the TI Programmable 58 with plug-in Solid State Software™ modules. A Master Library module with 25 pre-written programs, including math, statistics, and finance, is virtually a tool kit for today's professional. And there are optional modules available.

With the TI Programmable 59, you can use blank magnetic cards to record custom programs to run individually or with programs in the library modules. And with the TI Programmer, fast, accurate conversions and calculations can be performed in either hexadecimal, octal, or decimal number bases.

Want more? The PC-100A Printer, Plotter will turn your TI Programmable calculator into a quiet, high-speed printing calculator that prints, lists, and traces your program.





A Powerful, Personal S-100 Computer

The Vector 1 digital computer is a powerful, personal computer based on the 8080A or Z-80 microprocessor and the common S-100 bus structure. Vector 1 has 78 basic machine instructions and a minimum instruction cycle of two microseconds. There is room for up to 64 K of directly addressable memory using a parallel 8 bit word/16 bit address and up to 256 separate input and output devices can be addressed. For simplicity, reliability, and quality, see Vector 1 and the complete Vector Graphic line at ComputerLand.

The Quality Mini Recording Media

Your data is worthless if you can't store it safely until you need it. And not all brands of data storage media provide the safety you desire. Chances are you've already discovered that the hard way.

Reason enough to buy Verbatim diskettes and storage files at ComputerLand. Verbatim diskettes, cassettes, and cartridges store your data word-for-word until you call for it. The captured data is then played back bit for bit, byte for byte, verbatim. That saves your time. And your money. When you want to be sure of your data, specify Verbatim at ComputerLand.

You're always welcome at ComputerLand

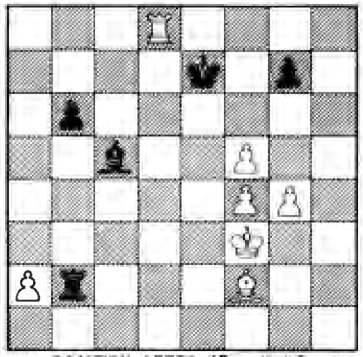
AL: Huntsville
AR: Little Rock
AZ: Phoenix
CA: Dublin, El Cerrito, Hayward, Inglewood, Lawndale, Los Altos, Mission Viejo, San Bernardino, San Diego, San Francisco, San Jose, San Mateo, Santa Rosa, Thousand Oaks, Tustin, Walnut Creek
CO: Colorado Springs, Denver
CT: Fairfield
DC: Washington
DE: Newark
FL: Boca Raton, Ft. Lauderdale
GA: Atlanta
HI: Honolulu
IL: Arlington Heights, Downers Grove, Niles, Oak Lawn, Peoria
IN: Indianapolis
KA: •Kansas City
KY: Louisville
MD: Rockville
MI: Detroit, Grand Rapids
MN: Minneapolis
MO: Springfield
NH: Nashua
NJ: Cherry Hill, Morristown, Paramus
NY: Buffalo, Ithaca, •Long Island
NC: Charlotte
OH: Cleveland, Columbus
OR: Portland
PA: Harrisburg
TN: Memphis
TX: Austin, Clear Lake, Dallas, Houston
WA: Bellevue, Federal Way, Tacoma
WI: Madison, Milwaukee
INT'L: Sydney, Australia; Brussels, Belgium; Toronto, Canada; Winnipeg, Canada

•Opening in December

ComputerLand® Corp., 14400 Catalina St., San Leandro, CA 94577 (415) 895-9363

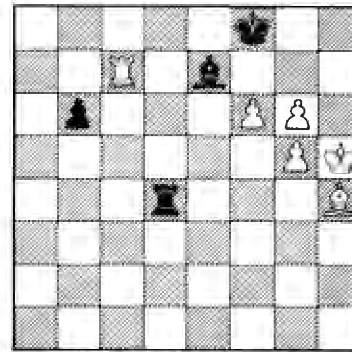
Circle 75 on inquiry card.

GAME 4



POSITION AFTER 47. . . K-K2

GAME 4



POSITION AFTER 56. P-B6

Figure 3: Position reached in game 4 after Black has made his 47th move. The human chess masters present, including Canadian Master Bruce Amos and 14 year old US National Master Joel Benjamin, were of the opinion that White must lose material. White did have a move they missed, and played it.

Figure 4: The final position of game 4. White's pawns will march irresistibly to the eighth rank and become Queens. Black can find no way to stop them, and resigns.

	Chess 4.7	Levy
1.	P-K4	P-K4
2.	N-KB3	P-KB4
3.	PxP	P-K5
4.	N-K5	N-KB3
5.	N-N4	P-Q4
6.	NxN check	QxN
7.	Q-R5 check	Q-B2
8.	QxQ check	KxQ
9.	N-B3	P-B3

At this point, Levy announced to the spectators that he was playing the "Smash-Crash" Gambit, attributed to Josef Smolij of Toronto.

10.	P-Q3	PxP
11.	BxP	

Possessing a one pawn advantage, the computer has forced Black's King to remain in the center of the board.

11.	. . .	N-Q2
12.	B-KB4	N-B4
13.	P-KN4	NxB check
14.	PxN	B-B4
15.	O-O	P-KR4
16.	N-R4	B-Q5
17.	B-K3	B-K4
18.	P-Q4	B-Q3
19.	P-KR3	P-QN3
20.	R/B-K1	B-Q2
21.	N-B3	PxP
22.	PxP	R-R5
23.	P-B3	R/1-R1

Levy has seized command of the King Rook file. The defense is not at all obvious.

24.	K-B1	B-N6
25.	R-K2	B-B1

The move 25. . . R-R8 with check fails because of the reply B-N1.

26.	K-N2	B-Q3
27.	B-N1	R-R6
28.	R/1-K1	R-N6 check
29.	K-B2	R/1-R6
30.	R-K3	B-R3
31.	N-K2	

Chess 4.7 forces the exchange of minor pieces, and thereby defangs Levy's attack.

31.	. . .	BxN
32.	R/1xB	P-B4
33.	P-B4	RxR
34.	RxR	R-R5
35.	K-N3	R-R8
36.	B-B2	R-Q8
37.	R-R3	PxP
38.	RxP check	K-B1
39.	R-Q7	R-Q6 check
40.	K-N2	B-B4
41.	RxP/5	R-Q7
42.	P-N4	BxP
43.	R-Q8 check	K-B2
44.	R-Q7 check	K-B1
45.	RxP/4	R-N7
46.	K-B3	

This move avoids the pin of the Bishop to the King — see why in the next move.

46.	. . .	B-B4
47.	R-Q8 check	K-K2
48.	B-R4! check	

The human masters present did not see this move. They thought the computer was certain to lose material.

48.	. . .	K-B2
49.	P-N5	P-N3
50.	R-Q7 check	K-B1
51.	PxP	RxP
52.	P-B5	R-R6 check
53.	K-N4	R-R5 check
54.	K-R5	R-Q5
55.	R-QB7	B-K2
56.	P-B6	

Black has no way to prevent the steamroller pawns from advancing to the eighth rank.

56.	. . .	Resigns.

Table 2: The score of the fourth round game. The computer had the White pieces and the first move. After Levy lost the game, Joe Smolij complained that the Smash-Crash Gambit was for use against people, not machines.

Photo 2: Josef Smolij, the guru of the Yonge Street Chess Association, as he presides over his midnight lessons in the Smash-Crash Gambit.

Photo 3: In game 4, David Levy stolidly ponders the position after his move 51 . . . . RxP. The computer's material and positional advantage is large, but tenaciously he seeks the best defense.

Photo 4: Levy forms his plan, and reaches out over the flickering electronic chessboard to put it into effect. He may persuade the computer to trade Rooks. Getting rid of Chess 4.7's troublesome Rook would allow some freedom of movement for Levy's beleaguered King.

Photo 5: The computer decides not to trade Rooks. Levy pulls his Bishop back to act as a shield against the final assault. He smiles as he sees that the steamroller pawns will not stop.

assembled experts had predicted a 3 game conclusion to the 6 game match. The rules required that Levy obtain only three points to win his wager. Now play would be forced to at least four rounds. Levy's concentration during the opening phase of the second game did not falter as he quietly put away the machine without apparent trouble.

Round 3 was not scheduled for six days, so the glass booth, looking much like an abandoned bus stop enclosure, sat empty while various masters played simultaneous exhibitions against spectators, amidst the sounds of three bowling lanes and the clank of sabers from the adjacent fencing matches.

Play resumed on September 2. The third round was another closed and quiet game which Levy won without apparent effort. The score then stood ½ to 2½, with Levy needing only a draw to win the match. However, he chose to confront Chess 4.7 directly in the fourth round by playing the Greco Counter Gambit. His decision was made only hours before, while sitting on a milk carton playing chess against Joe Smolij, the Smash-Crash Gambit expert.

Round 4 commenced with fireworks that never died out during the entire game. The moves of that game are given in table 2.

Though Levy finished the match in the fifth round with another closed game and held his 10 year wager, those on the computer chess side of the contest did demonstrate the ability to produce master level games. The most frequently heard comment after the match was that there were no losers in Toronto.

What happens now? A new version of the program, Chess 5.0, waits in the wings, the CYBER 176 spends most of its waking hours hard at work aiding in the design of its successor, and Levy has offered a prize of $5000 to the developer of a system which is able to defeat him in match play within the next five years. Here we go again.■

The answer to the Turing experiment question (page 86):

David Levy was playing White; Chess 4.7 was playing the Black pieces.

The computer found a surprising combination in game 1. The diagram of figure 1 shows the position immediately following Levy's move 12, P-R3. In the opinion of the computer operators, Chess 4.7 did not have any definite plan when it moved its Knight to the fifth rank. But Levy took 510 seconds to advance his Rook pawn. The program, calculating during all this time, explored enough move trees to find the hidden benefit in the otherwise unlikely appearing move of Knight takes King pawn.

The key move in the combination came after Black played its Queen to King Knight 4. Levy later said that if a human master had played the Knight sacrifice against him, he would have resigned immediately. As it was, he played on, confident that he could outmaneuver the machine in the endgame. His confidence was justified, and he managed to salvage a draw.

Photo 6: David Slate (left), of Northwestern University, and David Cahlander of Control Data Corp watch the computer terminal as it displays one of Chess 4.7's moves in game 4.

TRS-80 PET APPLE

PLAY CHESS WITH YOUR COMPUTER!

MICROCHESS is the culmination of two years of chessplaying program development by **Peter Jennings**, author of the famous 1K byte chess program for the KIM-1. MICROCHESS 2.0 for 8K PETs and 16K APPLEs, in 6502 machine language, offers 8 levels of play to suit everyone from the beginner learning chess to the serious player. It examines positions as many as 6 moves ahead, and includes a chess clock for tournament play. MICROCHESS 1.5 for 4K TRS-80s, in Z-80 machine language, offers 3 levels of play (both Level I and Level II versions are included and can be loaded on any TRS-80 without TBUG). MICROCHESS checks every move for legality and displays the current position on a graphic chessboard. You can play White or Black, set up and play from special board positions, or even watch the computer play against itself! Available now at a special introductory price of only **$19.95**

BRIDGE CHALLENGER by **George Duisman** for 8K PETs, Level II 16K TRS-80s, and 16K APPLEs: You and the dummy play 4 person Contract Bridge against the computer. The program will deal hands at random or according to your criterion for high card points. You can review tricks, swap sides or replay hands when the cards are known. No longer do you need 4 people to play! **$14.95**

STIMULATING SIMULATIONS by **Dr. C.W. Engel** for 8K PETs, 4K Level I and II TRS-80s, and APPLEs with Applesoft II: Ten original simulation games such as Forest Fire, Lost Treasure, Gone Fishing and Diamond Thief, progressing from elementary to quite complex with most suitable for schoolchildren. Includes a 64 page book giving flowcharts, listings and suggested modifications ... **$14.95**

ORDERS: *Check, money order or VISA/Master Charge accepted; programs and cassettes guaranteed. If you have questions, please call us at* ***617-783-0694****. If you know what you want and have your VISA/MC card ready, you can* ***DIAL TOLL FREE 1-800-325-6400*** *(24 hours, 7 days; in Missouri, dial 1-800-342-6600). Or you can mail your order to the address below. Personal Software™ products are now available at all Radio Shack® stores throughout the United States and Canada, and from the PET and APPLE dealers listed below. New dealers are being added daily. For the name and address of a dealer near you, call us at* ***617-783-0694*** *today!*

Radio Shack® theComputerStore™ ComputerLand®

ALASKA

ALPHA ELECTRONICS
601 E Northern Light Blvd
Anchorage, AK 99503

CALIFORNIA

ADVANCED COMPUTER PRODUCTS
1310 B East Edinger
Santa Ana, CA 92705

BYTE SHOP
87 Marina Center
Suisan, CA 94585

CAPITOL COMPUTER SYSTEMS
3396 El Camino Ave
Sacramento, CA 95616

MARIN COMPUTER CENTER
70 Skyview Terrace
San Raphael, CA 94903

RAINBOW COMPUTING
10723 White Oak Ave
Granada Hills, CA 91344

SMALL SYSTEM SOFTWARE
2178 Hood Dr
Thousand Oaks, CA 91360

WASHINGTON D.C.

COMPUTER CABLEVISION
2617 42nd St NW #2
Washington, D C 20007

FLORIDA

FOCUS SCIENTIFIC ENTERPRISES
1601 Biscayne Blvd
Miami, FL 33132

MICROCOMPUTER SYSTEMS
144 S Del Mabry Hwy
Tampa, FL 33609

ILLINOIS

EMMANUEL B GARCIA JR AND ASSOCIATES
3950 N Lake Shore Dr #2310
Chicago, IL 60613

MASSACHUSETTS

THE COMPUTER STORE, INC
120 Cambridge St
Burlington, MA 01803

NEW ENGLAND ELECTRONICS
248 Bridge St
Springfield, MA 01103

MICHIGAN

COMPUTERLAND
29673 Northwestern Hwy
Southfield, MI 48034

COMPUTER MART OF ROYAL OAK
1800 W Fourteen Mile Rd
Royal Oak, MI 48073

LEVEL FOUR PRODUCTIONS
32020 Cheboygan
Westland, MI 48185

NEW DIMENSIONS IN COMPUTING
541 E Grand River
East Lansing, MI 48823

NEWMAN COMPUTER EXCHANGE
1250 N Main St
Ann Arbor, MI 48104

MINNESOTA

COMPUTERLAND
8070 Morgan Circle Dr
Bloomington, MN 55431

NEW YORK

THE COMPUTER CORNER
White Plains Mall
200 Hamilton Ave
White Plains, NY 10601

THE COMPUTER FACTORY
485 Lexington Ave
New York, NY 10017

COMPUTER MICROSYSTEMS
1311 Northern Blvd
Manhasset, NY 11030

LONG ISLAND COMPUTER GENERAL STORE
103 Atlantic Ave
Lynbrook, NY 11563

NORTH CAROLINA

BYTE SHOP
218 N Elm St.
Greensboro, NC 27401

COMPUTERLAND
3915 E Independence
Charlotte, NC 28205

MICROCOMPUTER SERVICES
108-110 Arcade Bldg.
Hickory, NC 28601

OHIO

ASTRO VIDEO ELECTRONICS
504 E. Main St.
Lancaster, OH 43130

OKLAHOMA

BUSINESS MICRO SYSTEMS
7228 W. Reno, Rte 5
Oklahoma City, OK 73108

PENNSYLVANIA

A B COMPUTERS
c/o Innomark
1411 Callowhill
Perkasie, PA 18944

TENNESSEE

COMPUTER LABS OF MEMPHIS
627 S Mendenhall
Memphis, TN 38117

TEXAS

COMPUTER TERMINAL
2101 Myrtle
El Paso, TX 79901

UTAH

THE HI-FI SHOP
4680 Holladay Blvd
Salt Lake City, UT 84117

VIRGINIA

THE COMPUTER PLACE
2718 Colonial Ave
Roanoke, VA 24015

COMPUTERS PLUS
678 S. Pickett St
Alexandria, VA 22304

COMPUTER HARDWARE STORE
818 Franklin St
Alexandria, VA 22314

HOME COMPUTER CENTER
12588 Warwick Blvd
Newport News, VA 23606

WASHINGTON

YE OLDE COMPUTER SHOPPE
1301 George Washington
Richland, WA 99352

CANADA

COMPUTER INNOVATIONS
36 Beechmont Crescent
Ottawa, Ontario K1B 4A8

KOBETEK SYSTEMS
RR #1 Wolfville
Nova Scotia B0P 1X0

ENGLAND

BYTE SHOP
426-428 Cranbrooke Rd
Gants Hill, Ilford, Essex

PETSOFT SALES
P O Box 9
Newbury, Berkshire RG13 1PB

GERMANY

ING W HOFACKER GMBH
8 Munchen 75
Postfach 437

NETHERLANDS

COMPUTRON
P O Box 18663
Den Haag 2502 ER

P.O. Box 136-B12 **Personal Software™** Cambridge, MA 02138

Circle 302 on inquiry card.

Nybbles

Z-80 Assembler

Patrick A Crowe
22 Ringsbury Close
Purton
Swindon ENGLAND SN5 9DE

A Z-80 assembler that implements all of the Zilog defined mnemonics is available from BYTE. This assembler uses the conventions established by Zilog in the *Z-80 Assembly Language Programming Manual.* It recognizes uppercase characters for labels, operators and operands. All defined pseudo-operations have been implemented except for macroinstructions and conditional assembly commands.

The assembler can be implemented in read only memory. It assumes that a console display and a paper tape reader and punch are available. However, since the user must supply the input and output routines for the program, this is not necessary, and the required functions can point to locations in memory.

The Nybbles Library is an inexpensive means for BYTE readers to share some interesting but specialized forms of software. These programs are written by readers with small computers and printer facilities, and are therefore designed for particular systems. The algorithms and programming techniques in these programs can be directly used by readers with similar equipment, or can serve as an inspiration for improvisation on computers of different characteristics.

Potential authors of such programs should send us a self-addressed stamped envelope, with a request for a copy of our "Guidelines for Nybbles Authors." Payment for Nybbles items is based on sales and length of the item. Rates are set at the time of acceptance.

Nybbles Library programs are sent in listing form, printed on 8.5 by 11 inch paper on both sides. The Nybbles Library programs are 3 hole punched for collection in loose leaf binders, and come in an attractive folder which serves as a cover.

This month "The Z-80 Assembler" (#101) has been added to the Nybbles Library. To order your personal copy, at $4 postpaid, fill out the coupon below.

Please send_____copies of BYTE Nybble #_______ at $_________ postpaid.

_____*Check Enclosed*

*Bill my BAC #*______________________ *Exp Date*______________

*Bill my MC #*______________________ *Exp Date*______________

Name ____

Street ____

City____________________ **State** ________ **Zip Code** __________

Nybbles BYTE Nybbles Library, 70 Main St, Peterborough NH 03458

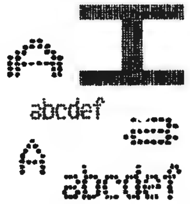
You may photocopy this page if you wish to keep your BYTE intact.

Pay a little bit more and get a printer that's brighter than your computer. The BrighterWriter.™

When a few dollars more buys you a first-class impact printer, why settle for a toy? The Brighter-Writer gives you quality to start with. And versatility that stays even if you outgrow your present personal computer.

Built smart like the big ones.

The BrighterWriter's a smart printer. There's a microcomputer inside. It outwits even the bigger, higher-priced printers. So you get versatility to do all kinds of printing. And power to grow on.



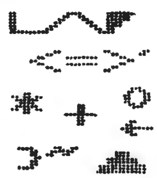
Prints fat, skinny, tall, small.*

This printer can be as creative as your imagination. Stretch out your characters. Squeeze them close. Make them high. Low. Bold. Banner. You name it.

Plugs into your computer.

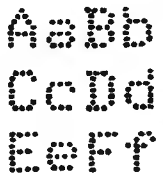
Most popular personal computers interface to the BrighterWriter. Simply and quickly. Hundreds of BrighterWriters are working in Apple, TRS-80, Heathkit, S-100 and many other personal computer systems right now.

Pictures and fancy symbols.*



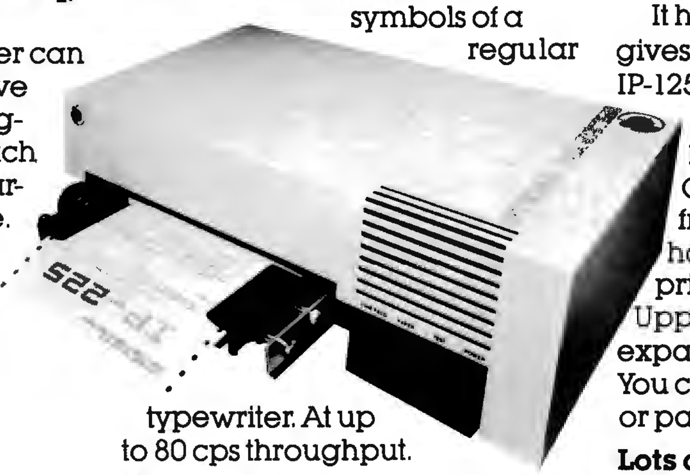
The BrighterWriter draws out your creativity. You can print drawings, graphs, diagrams, bold symbols, or just about any graphic you can imagine.

Picture your page as thousands of dots. The BrighterWriter can fill in the dots, plot them contiguously, stack them, or scatter them. And its special set of graphic characters simplifies the process.



Prints any character a typewriter can. Faster . . .

The BrighterWriter can print plain and simple. With 7x7 dot matrix clarity. You get all the letters, numbers, and standard symbols of a regular typewriter. At up to 80 cps throughput.



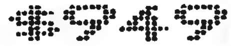
Ordinary paper.

Fancy or plain, the Brighter-Writer prints on ordinary paper. Better yet, it prints on many shapes of paper. Single sheets. Roll. Fanfold.

Want more copies? The BrighterWriter prints multiple copies without extra adjustments.

Four easy buttons.

Operating the BrighterWriter couldn't be simpler. Up-front controls are easy to get to. A power button to turn it on. A test button to self-test your printer. A paper feed button to advance the sheets or forms. A line feed button to advance the paper a line at a time.

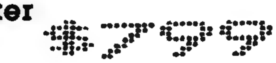


Prints any-which-way.

The BrighterWriter comes in two models. The IP-225, at $949, gives you a BrighterWriter with tractor-feed drive for precision forms control. This one can handle everything from labels to 8½" paper widths.

It has eight form lengths and gives you all the features of our IP-125.

A brighter buy.



Our IP-125, friction-feed, BrighterWriter has a 96 character set and prints on 8½" wide paper. Upper and lowercase. It prints expanded characters, too. You can choose a RS-232 serial or parallel interface. $799

Lots of goodies.

There's more. Choose all kinds of options for your BrighterWriter. Up to 132 characters per line, variable character densities, larger buffers, special graphics packages, interface cables, and more.

Give us a call or write. Integral Data Systems, 14 Tech Circle, Natick, MA 01760, (617) 237-7610.

Better yet, see the Brighter-Writer at the store nearest you.

Integral Data Systems, Inc.

*Some of these advantages require extra-cost options.

Circle 177 on inquiry card.

Interface Your Computer to a Printing Calculator

Robert H Astmann
58A Spring St
Red Bank NJ 07701

Hexadecimal Row Vector (RWVCT)	Hexadecimal Column Vector (CLVCT)	Key
08	01	"0"
04	01	"1"
04	02	"2"
04	04	"3"
02	01	"4"
02	02	"5"
02	04	"6"
01	01	"7"
01	02	"8"
01	04	"9"
0F	08	"T"
0F	04	Paper Advance
01	08	"#"
08	08	"+"

Table 1: Contents of the row vector and column vector tables (RWVCT and CLVCT, respectively) referenced by the program in listing 1.

There are many microprocessor applications in which it is desirable to produce a hard copy of numeric information being measured or computed, yet even the cheapest of today's low cost printers could easily be the most expensive component of such a system. A solution to this problem is to use one of the thermal printing calculators now available. By means of an interface to a microprocessor, the calculator integrated circuit can be given stimuli identical to those received during the normal pushing of the calculator keys. In this article I describe such an interface which was implemented using an Intel 8080A processor and a Texas Instruments 5050M printing electronic calculator.

Basic Control Procedure

I first describe the method by which data is normally entered on a calculator keypad. Referring to figure 1, each button on the keypad provides a unique connection between a column output line and a row input line. The calculator integrated circuit outputs a scan pulse to each column bus sequentially, and looks for an input pulse from one of the rows. The interpretation given to a detected row signal is therefore dependent on which column is being accessed during the given time period.

The job for the microprocessor in this application is to monitor the column signals until the column containing the desired character key is active and then to drive the correct row bus to a high level so that the calculator circuit senses the input while the given column signal is still active. The microprocessor software controls this procedure by using two stored lookup tables (see table 1): a list of column vector bytes and a list of row vector bytes. The entries in these two tables, together with the code for the

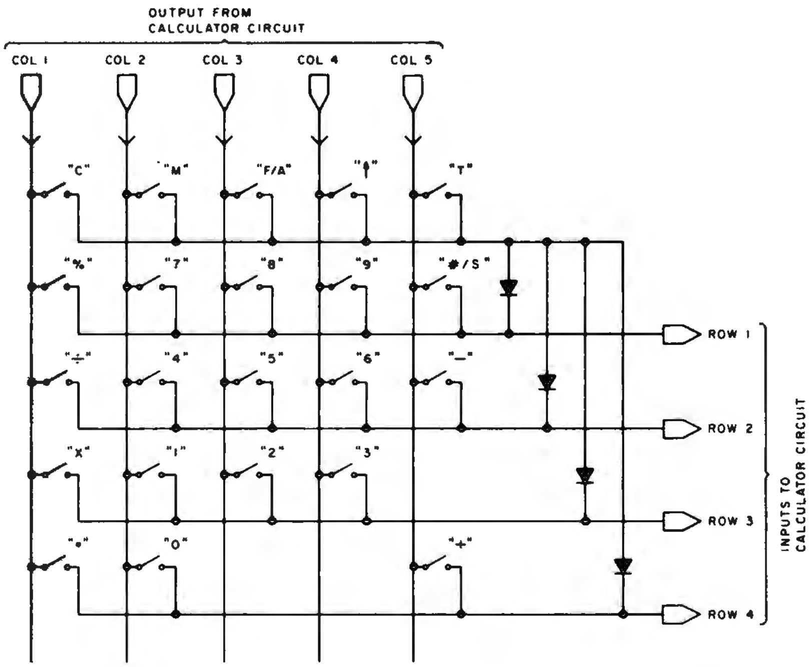


Figure 1: Keyboard arrangement of the Texas Instruments 5050M printing calculator. Calculator logic outputs a scan pulse to each column bus sequentially and looks for an input pulse from one of the rows. This uniquely identifies the key pressed by the user. Although there are only four row inputs to the calculator circuit, there are five rows of keys. Signals from the upper row of keys appear simultaneously on all four row inputs through the diode network.

COMING!

FRIDAY, SATURDAY, & SUNDAY, APRIL 6, 7, & 8, 1979

Be at the Northeast's most exciting show featuring microcomputers and small computer systems.

The Northeast Business & Personal Computer Show will be the largest presentation of hardware and software ever gathered in Boston. And it will all take place at the spectacular Hynes Auditorium in Boston's Prudential Center. Hundreds of displays and exhibits will showcase microcomputers and small computer systems for businesspeople ■ hobbyists ■ doctors ■ scientists ■ engineers ■ accountants ■ homeowners ■ researchers ■ programmers ■ technicians ■ students ■ educators. With special exhibits for children, featuring calculators, computers, and educational displays.

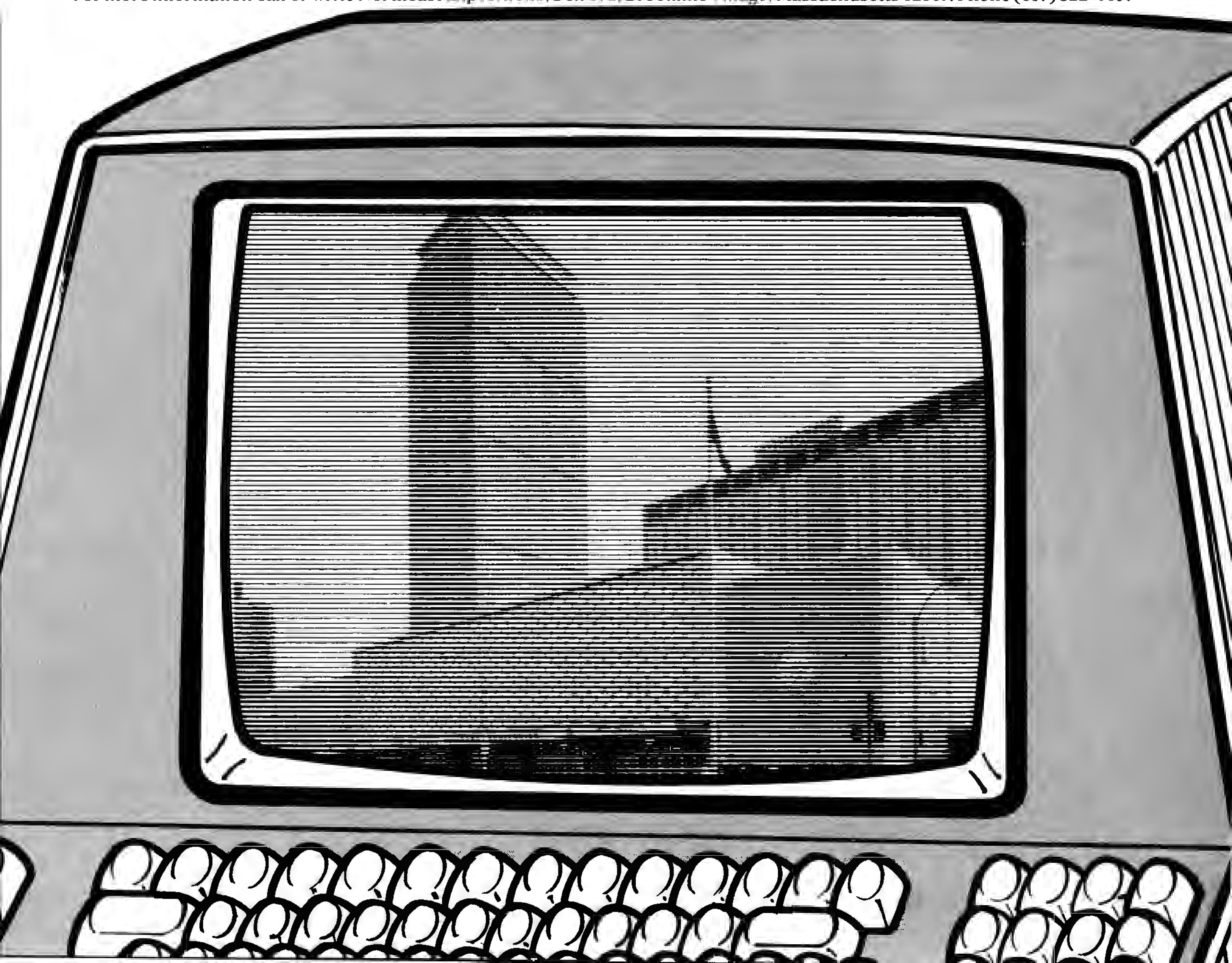
NORTHEAST PERSONAL & BUSINESS COMPUTER SHOW

You'll be entertained, educated, enthralled! Win valuable prizes in the computerized mouse maze contest, programming contests, and other electronic competitions. Hear computerized music synthesizers. See computer-generated art, graphics, and animation. Watch computer amusements. Play dozens of electronic and video games.

Internationally recognized speakers will give lectures and seminars for all categories and levels of enthusiasts, including introductory classes for novices. You may even have a chance to demonstrate your own equipment and applications. So plan to be in Boston this April. This is one show you won't want to miss!

HYNES AUDITORIUM, PRUDENTIAL CENTER, BOSTON

For more information call or write Northeast Expositions, Box 678, Brookline Village, Massachusetts 02147. Phone (617) 522-4467



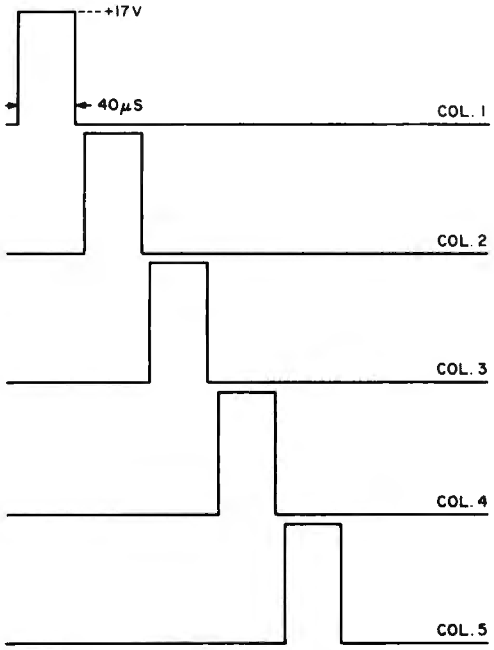
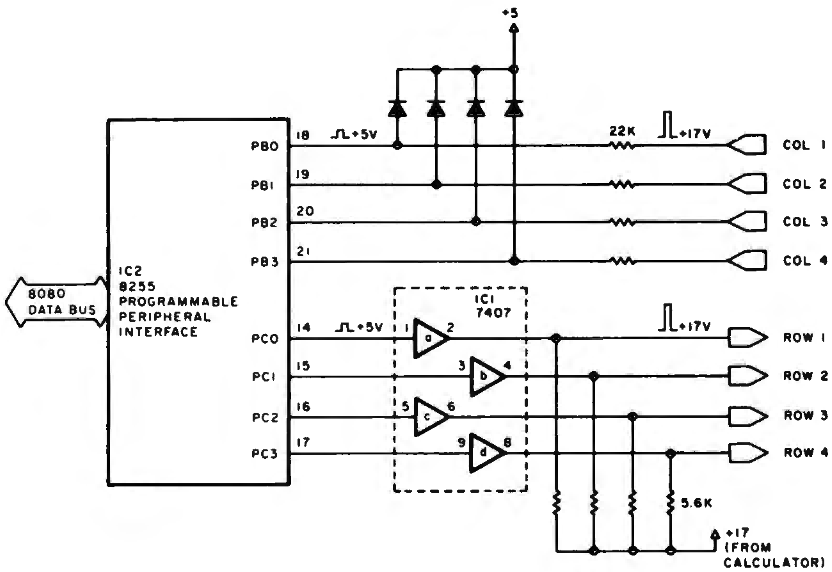


Figure 2: The sequence of column scan signals outputted by the calculator circuit.



POWER WIRING TABLE

NUMBER	TYPE	+5	GROUND
IC1	7407	14	7
IC2	8255	26	7

Figure 3: Circuitry for interfacing the 8255 programmable peripheral interface to the Texas Instruments 5050M printing calculator. The diodes prevent the inputs to the 8255 from going above 5.7 V when a column signal goes to 17 V. The output lines from the interface are connected to open collector drivers (IC1) which translate the 5 V signals to 17 V.

desired character, determine the mapping of a column input into a row output.

Hardware Interface

The signalling between the TI 5050M calculator integrated circuit and keypad is illustrated in figure 2. A 17 V pulse of 40 μs nominal duration is outputted to each column bus with the entire sequence being repeated every 7.3 ms. This signalling continues as long as no button is pushed. When a button is pushed, the column signals are extended to about 150 μs to validate the button push. This pulse width is maintained until the button is released, during which time any other button push is ignored. It is immediately apparent that to interface this calculator to the 8080A processor, level translation circuitry in both directions is required. The circuitry I used in this application is shown in figure 3. An Intel 8255 Programmable Peripheral Interface integrated circuit was chosen because it was already interfaced to the 8080A as a keyboard port. The diodes tied to +5 V insure that the inputs to the 8255 do not go above +5.7 V when a column signal goes to +17 V. The output lines from the 8255 are connected to open collector drivers which translate a +5 V signal to +17 V.

The necessary connection points within the calculator case were easily accessed since there were large metal strips connecting the printed circuit board to the keypad. A dual trace oscilloscope was used to deduce the identity of each connection. Once the columns were identified by noting the time displacement of the scan pulse on each bus relative to the others, the rows were identified by pressing buttons and looking for responses on the connections that normally did not exhibit any signalling.

Software Interface

At the heart of the printer control software is the driver program, INTER. INTER is called when a single decimal digit or control character is to be entered. A 4 bit BCD representation of the digit must first be loaded into the C register of the 8080A. As shown in listing 1, this character code is used to select the correct column and row vectors from the two lookup tables contained in table 1.

In my microcomputer system, which is based on an Altair 8800 computer, this program was executed out of programmable memory which utilizes a processor wait

ELEPHANT MEMORY

for peanuts!

At last, you can get a mammoth 32K memory without paying a beastly price. The SuperRam™ 32K **static** memory kit is just $649 for S-100 systems (Z-80 version, add $70).

The latest of George Morrow's heavyweight memory designs for Thinker Toys™, the SuperRam™ 32K features two independent 16K blocks, each addressable and write-protectable. It meets the Proposed IEEE S-100 Standard with full buffering of both data and address lines. Uncommonly efficient, the Morrow design utilizes just 7 IC's for control and buffering, drawing typically 2.6 amps.

Ask your local computer shop to order the SuperRam™ 32K kit for you. Or, if unavailable locally, order direct from Thinker Toys™, 1201 10th St., Berkeley, CA 94710. Or call (415) 524-2101, weekdays 10-5 Pacific Time. Add $3 for handling. (Cal. res. add tax.)

Morrow makes memory for

Thinker Toys™

Circle 255 on inquiry card.

Judy the elephant courtesy Marine World Africa USA state on each memory access. This had the effect of reducing program execution speed to the point where it was not possible to determine when the desired column was active and thus to output the appropriate row signal before the termination of the same 40 μs column signal pulse. Instead, INTER actually scans the column inputs until the column to the immediate left of the desired column is sensed active. Then there is a software delay, after which the correct row output is activated. Next there is a second software delay, after which all row outputs are zeroed. The initial values for the software delay counters were determined experimentally (ie: they were adjusted until the calculator consistently printed a character each time INTER was called).

The entries in the column vector table are set up so that a 01 vector will recognize column 1, 02 will recognize column 2, etc. Since INTER looks for the column to the immediate left of the column containing the desired character, characters in column 1 (ie: C, %, x, etc), cannot be printed unless INTER is modified to enable it to recognize when column 5 is active, then to delay for about 7.1 ms (ie: until column 1 is active) before outputting a row signal.

Just above INTER in the software hierarchy is PR2DI, the routine which enters two decimal digits. The source listing shows that INTER is called twice by PR2DI. Each call to INTER is followed by a software delay to allow the calculator chip to store the entered digit and return to the scan mode. (The digits are not printed, since the complete line of digits has not yet been terminated.) The two desired digits are passed to PR2DI in BCD form via the accumulator.

In my application, the largest number to be printed is four digits long. Hence the routine OUTPR enters four decimal digits by calling PR2DI twice. The BCD values for the four digits are passed to OUTPR via the BC register pair.

Finally, a segment of the main application program is shown. A series of 4-digit numbers is printed as they are fetched from memory. These numbers are already stored in BCD form in memory since they are the result of arithmetic operations that were followed by the 8080's decimal adjust instruction. First, the BC register pair is loaded with the codes for the four decimal digits. The memory data pointer must then be saved in scratchpad programmable memory since the HL register pair will be used by INTER. Next, OUTPR is called to transfer the four digits to the calculator. Following this operation, the calculator is waiting for either more digits or a line termination. The 5050M can be made to terminate and print a line by means of an operation keystroke or by using the #/S key. (If this key is pressed after an operation, it is interpreted as a request to print a subtotal. If pressed after a series of digits have been entered, the numbers are printed and the line is terminated with a "#". These numbers are ignored by the calculator, but serve as an index for the benefit of the

```
            Op
Label       Code   Operand     Commentary

            LHLD   POINT       ; load HL with memory data pointer
            MOV    C,M         ; load C from memory with 2 low order digits
            INX    H
Segment     MOV    B,M         ; load B from memory with 2 high order digits
of          INX    H
Main        SHLD   POINT       ; save memory data pointer
Program:    CALL   OUTPR       ; enter the four digits
            MVI    C,0CH       ; "#"
            CALL   LINE        ; terminate the line
            CALL   SKIP        ; skip a line
-------------------------------------------------------------------------------
OUTPR:      MOV    A,C         ; get 2 low order decimal digits
            STA    SAVE+1      ; save
            MOV    A,B         ; get 2 high order decimal digits
            MVI    B,0         ; clear B reg for INTER
            CALL   PR2DI       ; enter 2 high order digits
            LDA    SAVE+1      ; retrieve 2 low order digits
            JMP    PR2DI       ; enter 2 low order digits
-------------------------------------------------------------------------------
PR2DI:      STA    SAVE        ; save low order digit
            RRC
            RRC
            RRC
            RRC                ; BCD value for digit to be entered now
                                 occupies right side of accumulator
            ANI    0FH         ; clear left side of accumulator
            MOV    C,A
            CALL   INTER       ; enter the high order digit
            LXI    D,0F000H
            CALL   DELAY       ; delay for calculator response time
            LDA    SAVE        ; retrieve low order digit
            ANI    0FH
            MOV    C,A
            CALL   INTER       ; enter the low order digit
            LXI    D,0F000H
            JMP    DELAY
-------------------------------------------------------------------------------
SKIP:       MVI    C,0BH       ; paper advance
LINE:       CALL   INTER
            LXI    D,0
            JMP    DELAY       ; delay for thermal print head response and paper advance
-------------------------------------------------------------------------------
INTER:      LXI    H,CLVCT     ; HL points to head of column vector table
            DAD    B           ; HL points to correct column vector byte
            MOV    D,M         ; load D reg. with column vector
            LXI    H,RWVCT     ; HL points to head of row vector table
            DAD    B           ; HL points to correct row vector byte
COL:        IN     PORTB       ; read status of column signals
            ANA    D           ; is desired column active?
            JZ     COL         ; No, keep looking
            MOV    A,M         ; Yes, prepare to output row signal
            MVI    C,0FCH      ; initialize first delay counter
WAIT1       INR    C
            JNZ    WAIT1
            OUT    PORTC       ; time to output row signals
            MVI    C,0F0H      ; initialize second delay counter
WAIT2:      INR    C
            JNZ    WAIT2
            XRA    A           ; clear accumulator
            OUT    PORTC       ; reset row signals
            RET
-------------------------------------------------------------------------------
DELAY:      INR    E
            JNZ    DELAY
            INR    D
            JNZ    DELAY
            RET
```

Listing 1: Assembly language program for interfacing an Intel 8080A processor to a Texas Instruments 5050M printing calculator using an 8255 programmable peripheral interface. Row vector and column vector (RWVCT and CLVCT) contents are listed in table 1. The first section in the list is the portion of the main program that calls the routines.

user.) Hence, at this point in the application program the code for the #/S key is loaded into the C register and the subroutine LINE is called. LINE proceeds to call INTER and then causes a delay of about 0.5 seconds which enables the calculator to activate the thermal print heads and advance the paper. Finally, the subroutine SKIP is called in order to skip a line before printing the next number.

Conclusions

When using a microprocessor in a control application, it is necessary to be able to "shake hands" with the device to be controlled. This is best accomplished by structuring the software so that a low-level driver routine makes the handshaking transparent to the higher level software.

In this application, the signalling protocol of the printing calculator is utilized as a control mechanism by the processor. The interface between the two devices is easy to implement and the result is low cost numeric printing capability. Whenever the calculator is disconnected from the processor interface, it will operate normally again.■

Circle 291 on inquiry card.

Zapper

A Computer Driven EROM Programmer

G H Gable
50 Cliftwood Dr
Halesite NY 11743

One of the most fascinating and useful products of recent technology is the read only memory (often abbreviated as ROM) and especially useful for the experimental systems designer is the erasable and electrically programmable read only memory, variously abbreviated EROM or EPROM.

In designing my first microprocessor based system, a read only memory was a must to contain the operating system and the floating point arithmetic firmware. I did extensive research into read only memory systems and after a week or so I was ready to make a specification. I had previously chosen the processor for the system to be the MOS Technology 6502 which requires a memory access time of about 500 ns when running with a 1 MHz clock. It was very desirable to have the read only memory meet this specification for two reasons. First, because of the dynamic nature of the 6502, it does not wait for slow memory very readily. Second, and by far most important, I wanted my arithmetic routines in read only memory to run as fast as possible since I would be using them very often. These considerations ruled out the older 1702 type memories as too slow.

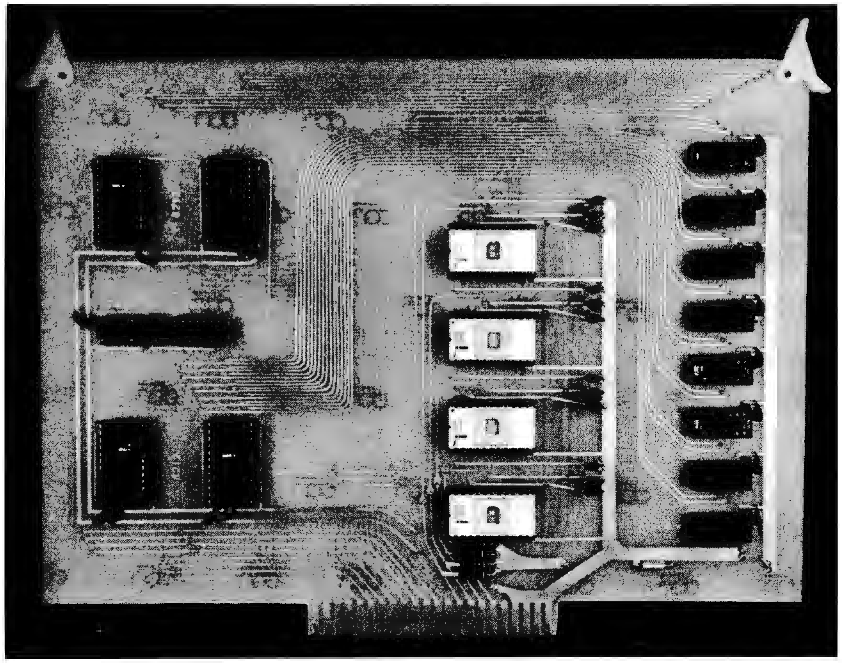


Photo 1: The prototype computer read only memory board with four EROM parts in the center. The eight sockets on the right hold the 1 K bytes of programmable memory which is selected by a jumper on the pins in the lower center of the board (off pin 24 of the lowest EROM). The 8212s on the left are the address bus drivers (lower pair). 8212s are abundantly used because they are the author's favorite all purpose medium scale integration chip.

The choice was obvious as soon as I read about the Intel 2708. It had all the requisite features: fast (450 ns) access time, large array (1024 8 bit words) on a single chip, and easy straightforward programming. When I designed this programmer the going price was $100; currently the prices have dropped to about $10, making this chip even more desirable.

The chip is also numbered 8708 to fit into Intel's 8000 line which includes the 8080. The 2708 and the 8708 are identical as far as I know. They are definitely interchangeable at a pin level. There is also a variation of the design called the 2704/8704 which is arranged as an array of 512 8 bit words. The 2704/8704 is electrically and logically identical to the 2708/8708 but contains only half as much memory. The high order address line is not defined for the 2704/8704. (Rumor has it that 2704/8704 parts are identical to 2708s but wired into the package with the high order address bit unconnected.)

System Design

My design called for 4 K bytes of read only memory resident firmware which could be built up over a period of time as the operating system and arithmetic routines were debugged. My approach to this was to prototype the eventual firmware in normal programmable memory and then transfer it to read only memory after debugging. I designed a 4 K byte read only memory board (photo 1) which has four 2708 PROM chips plus 1 K bytes of programmable memory. The programmable memory can be jumper selected to occupy any 1 K page on the board. This allows for prototyping a routine in the actual address space that it will eventually occupy. The system has worked out extremely well.

It was my original intention to have the read only memory programmed by professionals offsite. My impression was that 2708 programming was somewhat complex and that a programmer board for a limited number of burns was not very practical. After learning more about the 2708 my attitude changed. A little thought convinced me that a computer driven programmer could be simply constructed at minimum cost. It would be very convenient to be able to program the chips in my own computer and to be able to make changes and corrections with a short turnaround time.

Programming the EROM

When initially received, and after each erasure, all the bits of the 2708 are in the "1" state (output high). The content of the 2708 is programmed by selectively changing state to "0" in the desired bit locations. Programming a given byte requires the address of the byte on the address input pins and the data byte on the data pins, all at TTL levels (+5 V) with the write enable pin held at +12 V, a program pulse of +26 V at 20 mA is applied to the program pin. The 2708 specifications require that the program pulse be between 100 μs and 1000 μs wide. A series of pulses are required to program a particular address. Intel recommends that one pulse be administered to each address location in a loop. The number of times the loop must be repeated is a function of the pulse width. The final accumulated program current time to each address must be greater than 100 ms. Such a scheme is a natural for computer control.

The Zapper programming board shown in photo 2 and figure 1 is designed to have the address and data multiplexed to it through a peripheral interface adapter (PIA) with at least eleven output lines. I use the peripheral interface adapter that is available on my MOS Technology KIM-1 single board computer to drive the Zapper. If you do not have one of these PIAs I recommend either the MOS Technology 6520 or the Motorola 6820. The address and data are passed through the lower eight lines (PA0-PA7) while three of the upper lines (PB0-PB2) control the multiplexing and programming current.

The driving computer is expected to direct the following sequence of events which will program one address location in the 2708:

- PB0 is brought high to enable the upper 8212 (IC1) eight bit latch.
- The lower eight bits of the address are loaded on PA0-PA7 and thus into the 8212.
- PB0 is brought low latching the low address onto the outputs of IC1 which are wired to the address inputs of the 2708.
- PB1 is brought high to enable the lower 8212 (IC2). The upper two bits of the address are loaded on PA0-PA1 and latched when PB1 goes low.
- The data byte is loaded on PA0-PA7 and latched by the PIA.
- PB2 is brought high for the pulse time gating the program current to the EROM.

This sequence is repeated the required number of times to program the EROM.

A Note About UV Bulbs:

MSC Macalaster is located at Rte 111 and Everett Tpk, Nashua NH 03060.

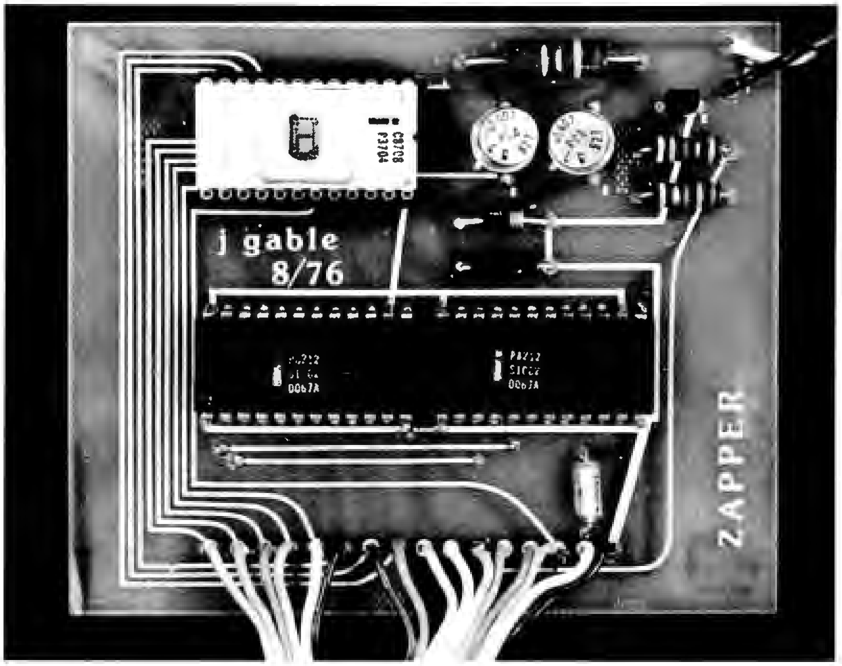


Photo 2: The Zapper board with the EROM in the upper left corner. Data from the PIA as well as logic power and ground come in via the ribbon cable at the bottom which is connected directly to the computer. Program power comes in on the cable in the upper right corner from an external power supply.

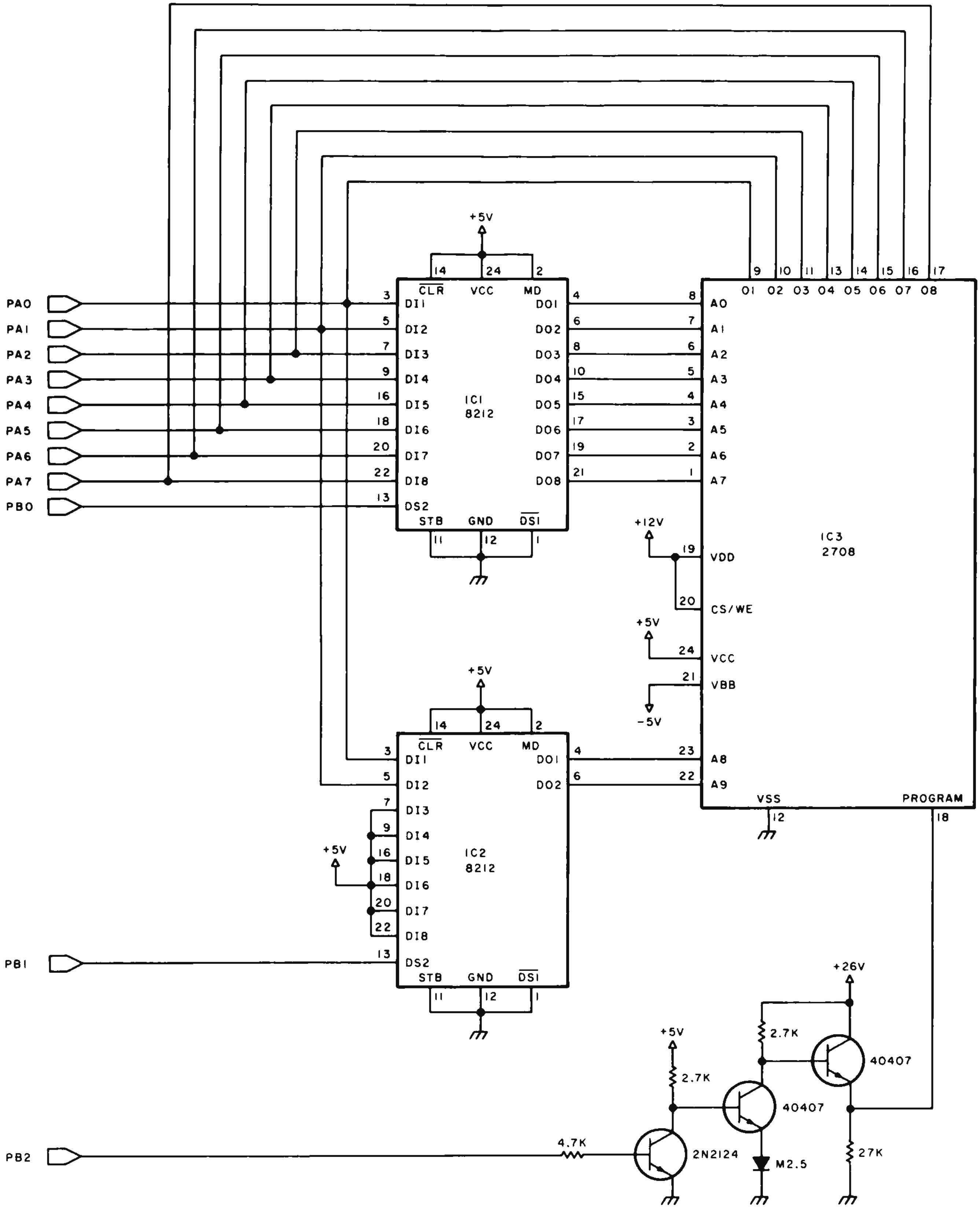


Figure 1: The address and data information for the Zapper is multiplexed through the PIA ports PA0-PA7 while control signals are presented on PB0-PB2. PB0 is connected to the enable pin of the upper 8212 which latches the lower eight bits of the address. The high two bits of the address are loaded and latched on the lower 8212 by PB1. The data byte is latched by the PIA. When PB2 goes high, program power is gated to the program pin of the 2708 by the 3 transistor high current gate in the lower right.

PRINTERS Printers PRINTERS Printers PRINTERS Printers

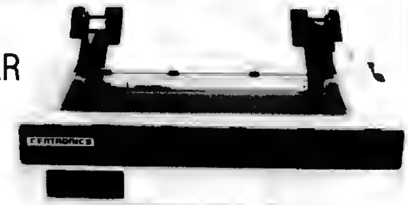
FROM SYNCHRO-SOUND THE COMPUTER PEOPLE AT PRICES YOU CAN'T AFFORD TO IGNORE!

OKIDATA SL 125
LINE PRINTER

- 125 lines per minute. 132-column print line
- Upper/lower case
- 8 different character sizes
- 12 IPS paper
- Tractor feed

$84.35 per month*
$2449.00 complete

CENTRONICS 703
SERIAL PRINTER



- 180 characters per second
- Bidirectional logic seeking printing
- Microprocessor electronics
- Excellent print quality
- Full 132 columns

$96.61 per month* **$2805.00 complete**

OKIDATA MODEL 110
LINE PRINTER

- 110 CPS dot matrix
- Friction feed
- 80 columns
- Bidirectional printing

$41.29 per month* **$1199.00 complete**

CENTRONICS 761 KSR
PRINTER

- 300 Baud serial transmission
- Bidirectional and incremental printing
- RS232, CCITT-V24 or current loop interface
- Baud selection (110/150/300)

$69.75 per month*
$2025.00 complete

TELETYPE MODEL 43
PRINTER

- 132 characters per line
- 110 or 300 Baud switch selectable
- Full keyboard
- RS232C Serial interface standard

$44.24 per month* **$1299.00 complete**

MORE PRINTER BUYS

	per month	complete
DECWRITER II LA 36	$49.94	$1495.00
LEAR-SIEGLER 301 PRINTER	65.27	1995.00
CENTRONICS 700 PRINTER	52.35	1520.00
CENTRONICS 779 PRINTER Tractor Feed	41.95	1294.00

Call for prices and information on other models in our complete line of printers and other hardware.

SYNCHRO-SOUND ENTERPRISES, INC.

The Computer People
193-25 Jamaica Avenue, Jamaica, New York 11423
212/468-7067 TWX 710-582-5886
West Coast: 5810 Commerce Blvd., Rohnert Park, CA 94928
707/544-2865

*36 month lease
Maintenance additional

OEM pricing available

Hours 9-4 Daily and Saturday

Visit our new showroom
Working units on display

BankAmericard Master Charge

```
                        ;       PROGRAM ZAPPER
                        ;
                        ;       THIS LISTING WAS PREPARED
                        ;       FROM A HAND ASSEMBLED SOURCE
                        ;
0200  A9  FF            ZAP     LDA  $FF       INITIALIZE CYCLE
0202  85  05                    STA  CYR       COUNT TO 255
0204  8D  01  17                STA  PADD      SET PIA DIRECTION
0207  A9  7F                    LDA  $7F       REGISTERS FOR
0209  8D  03  17                STA  PBDD      OUTPUT PORTS
020C  A9  00                    LDA  $00       SET CONTROL PORTS
020E  8D  02  17                STA  PBD       TO "OFF"
0211  20  71  02                JSR  MSG       WAIT FOR PROG POWER
0214  A2  00                    LDX  $00       SET INDIRECT INDEX
0216  A5  00            NXCY    LDA  BSL       TRANSFER STARTING
0218  85  10                    STA  LRL       ADDRESS TO LOCATION
021A  A5  01                    LDA  BSH       REGISTER
021C  85  11                    STA  LRH
021E  20  3F  02        NXAD    JSR  BURN      BURN PULSE
0221  E6  10                    INC  LRL       INCREMENT
0223  D0  02                    BNE  A1        LOCATION REGISTER
0225  E6  11                    INC  LRH
0227  A5  10            A1      LDA  LRL
0229  C5  02                    CMP  BEL       COMPARE LOCATION
022B  D0  F1                    BNE  NXAD      REGISTER WITH END
022D  A5  11                    LDA  LRH       ADDRESS TO CHECK
022F  C5  03                    CMP  BEH       FOR END OF BLOCK
0231  D0  EB                    BNE  NXAD
0233  C6  05                    DEC  CYR       DECREMENT CYCLE
0235  D0  D5                    BNE  NXCY      COUNT
0237  A9  07                    LDA  $07       RING TTY BELL
0239  20  A0  1E                JSR  OUTCH     WHEN DONE AND
023C  4C  4F  1C                JMP  MONITOR   RETURN TO MONITOR
023F
```

Listing 1: The Zapper program programmable memory starting address (BSL,BSH) and ending address plus one (BEL,BEH) are set before execution. The driving program sets up the PIA ports as outputs and insures that the control lines (PBD) are off (zeros) before programming power is applied. The loop through the addresses in the location register (LRL,LRH) supplies a burn pulse for each location. The cycle is repeated so that each location receives 255 pulses. The end of the program is signaled by the Teletype bell or terminal signal.

```
                        ;
                        ;       PROGRAM ZAPPER/BURN
                        ;
023F  A9  01            BURN    LDA  $01       OPEN LOW ADDRESS
0241  8D  02  17                STA  PBD       BUFFER
0244  A5  10                    LDA  LRL       GET LOW ADDRESS AND
0246  8D  00  17                STA  PAD       PUT IN BUFFER
0249  A9  02                    LDA  $02       OPEN HIGH ADDRESS
024B  8D  02  17                STA  PBD       BUFFER
024E  A5  11                    LDA  LRH       GET HIGH ADDRESS AND
0250  8D  00  17                STA  PAD       PUT IN BUFFER
0253  A9  00                    LDA  $00       LATCH ADDRESS
0255  8D  02  17                STA  PBD       BUFFERS
0258  A1  10                    LDA  LRL(IX)   GET DATA AND
025A  8D  00  17                STA  PAD       LATCH
025D  E6  04                    INC  WR        WAIT FOR DATA TO SETTLE
025F  A9  04                    LDA  $04       TURN ON PROGRAM
0261  8D  02  17                STA  PBD       POWER
0264  A0  43                    LDY  $43       WAIT 600US
0266  E6  04            AGN     INC  WR
0268  88                        DEY
0269  D0  FB                    BNE  AGN
026B  A9  00                    LDA  $00       TURN OFF PROGRAM
026D  8D  02  17                STA  PBD       POWER
0270  60                        RTS            RETURN
0271
```

Listing 2: The burn subroutine multiplexes the address and data through the PIA port (PAD) controlled by the control lines (PBD). A 600 μs programming pulse is applied after the address and data have been latched. The INC WR instruction does nothing more than provide a 5 μs delay. It is used first to let the data lines to the EROM settle before the programming pulse is applied. Later it is used in the pulse timing loop simply to cut down the number of iterations.

Software

The driving software, as shown in listings 1 to 4, implements the above sequence of events in a double loop. The inside loop, listing 2, works its way through all the addresses to be programmed and gives each location a 600 μs programming pulse. The outer loop, listing 1, repeats the process 255 times giving a total program current time of 153 ms to each bit. This is sufficient time to program the 2708. The start and end plus one addresses of the programmable memory block are loaded in BSL, BSH and BEL, BEH registers respectively before execution is begun. Data is programmed into the same relative addresses in the read only memory as they are found in the programmable memory; ie: the low ten bits of the address are the same.

Notice that the 2708 can be partially programmed. If the memory block to be copied is less than 1024 bytes long, only the appropriate bytes are programmed. The remaining locations are unchanged. The block to be programmed can start and end anywhere in the 1 K page. This is a very useful feature as it allows firmware to be developed over a period of time. The partially programmed read only memory can be used in the meantime. Incidentally, listings 2 and 3 are subroutines only for the sake of modularity and the whim of the author. They are called at only one point each.

It is very important that the +26 V programming power be off at the power supply until the computer has had a chance to latch PB2 low. After this initialization, a pause is built in to allow the operator to turn on the power supply before continuing. This pause is implemented by waiting for input from a terminal in subroutine MSG, listing 3. The application of program power before the computer has initialized the Zapper board will usually result in some random location being burned with some random data.

Erasing the EROM

The 2708 is very easily erased using an ultraviolet light source. Intel specifications indicate that an integrated dose of 10 watt-sec/cm^2 at a wavelength of 2537 angstroms is required to erase the 2708. A quick glance at the *CRC Handbook of Chemistry and Physics* shows that 2537 angstroms is the most persistent spectral line of mercury (Hg). This means that any mercury vapor lamp will do the trick. I use a nice packaged source from MSC Macalaster (Catalog #3400) which slips over the top of the read only memory. (When using the unit, discard the filters which come with it, and be sure you shield your eyes from the lamp.) The chip

```
                           ;
                           ;       PROGRAM ZAPPER/MSG
                           ;
0271   A2   00             MSG     LDX  $00        OUTPUT THE MESSSAGE
0273   BD   BI   02        MI      LDA  SMG+X      FROM DATA BLOCK
0276   C9   00                     CMP  $00        (SMG) UNTIL "NUL"
0278   F0   07                     BEQ  RET        CHARACTER FOUND
027A   20   A0   1E                JSR  OUTCH
027D   E8                          INX
027E   4C   73   02                JMP  MI
0281   20   5A   1E        RET     JSR  GETCH      WAIT FOR KEYSTROKE
0284   60                          RTS             BEFORE RETURNING

0285   0D   0A   0A        SMG     DATA             CR/LF/LF
0288   54   55   52                                 "TURN ON 26V--
028B   4E   20   4F                                 PUSH ANY KEY"
028E   4E   20   32                                 CR/LF/LF/NUL
0292   36   56   20
0295   2D   2D   20
0298   50   55   53
029B   48   20   41
029E   4E   59   20
02A2   4B   45   59
02A5   0D   0A   0A
02A8   00
02A9
```

Listing 3: The MSG routine effectively causes a pause so that programming power may be turned on after the Zapper board has been initialized. Execution is resumed when any key on the Teletype is pressed.

HEXADECIMAL LOCATION	SYMBOL	COMMENTS
0000	BSL	STARTING ADDRESS OF
0001	BSH	PROGRAMMABLE MEMORY.
0002	BEL	ENDING ADDRESS PLUS ONE OF
0003	BEH	PROGRAMMABLE MEMORY.
0004	WR	WAIT REGISTER.
0005	CYR	CYCLE COUNT REGISTER.
0010	LRL	LOCATION REGISTER.
0011	LRH	
1700	PAD	PA PORT DATA REGISTER.
1701	PADD	PA PORT DIRECTION REGISTER.
1702	PBD	PB PORT DATA REGISTER.
1703	PBDD	PB PORT DIRECTION REGISTER.
1EA0	OUTCH	TTY OUTPUT ROUTINE.
1F5A	GETCH	TTY INPUT ROUTINE.

Listing 4: External symbol table. The PIA registers (PAD, PADD, PBD, PBDD) are those assigned on the KIM-1 board. OUTCH and GETCH respectively output and input one character each to or from a terminal. They are part of the KIM-1 operating system.

should be stuck in a piece of conducting foam while erasing. An exposure time of 30 to 40 minutes will yield a fresh chip ready to be programmed again. If you want to make your own eraser, use the GE #G4S11 4 W mercury vapor lamp with a GE #89C504 ballast. Both of these items are usually available at commercial electrical supply houses. The exposure time is about 40 minutes with the 2708 placed 1 cm from the bulb.

My experience shows that each successive time a 2708 is erased the exposure time to completely erase it increases. As the total energy needed to erase it is cumulative, extra short exposures can be given as needed. A little program to check each byte for all ones will assure that the memory is fully erased.

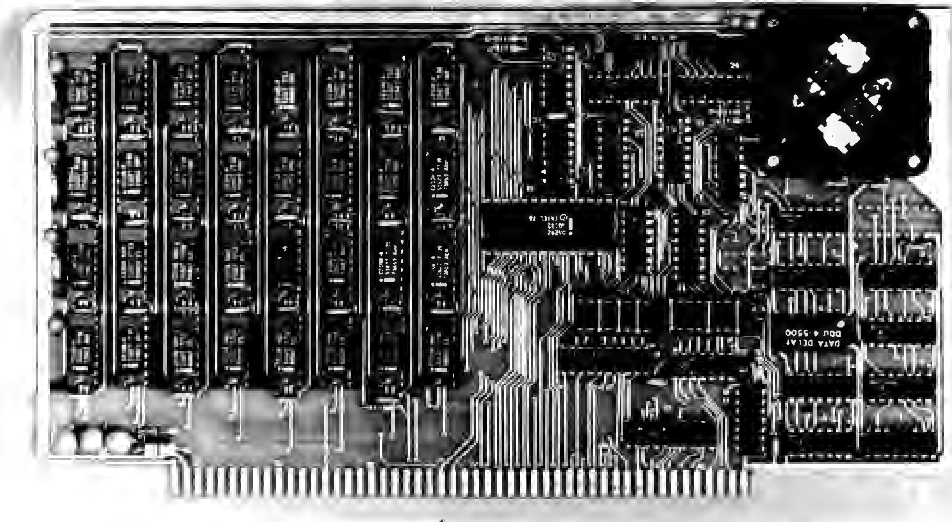
It is also convenient to remember that any 1 bit in the EROM can be changed to 0. Sometimes a single byte needs to be modified and this can occasionally be done without erasing the EROM and reprogramming it. This has been the case for me more often then statistics would dictate. Someone else must not be nearly so lucky.■

REFERENCES

1. *Intel Data Catalog,* Intel Corp, 1975.
2. *Memory Design Handbook*, Intel Corp, 1975.
3. *8080 Microcomputer Systems User's Manual,* Intel Corp, September 1975.

Intel Corporation
Literature Dept
3065 Bowers Av
Santa Clara CA 95051

Circle 303 on inquiry card.



We Start With A Price That's Hard To Beat.

But We Don't Stop There.

Central Data Corporation has combined the benefits of new technology and high-volume company sales to bring you a RAM board with more features and product options for less money.

Lower Prices
More Memory Capability

To begin with, we've reduced the price of our 16K RAM board by $40 to $249. At $425—a price reduction of $50—our 32K board costs less, too. Plus, we now offer a full 48K memory board for $599. These boards are expandable to 64K at a price of $185 per 16K package. Or you can start right out with a full 64K board for $775.

Improved Board Design

We've also added improvements to the board design at no extra cost to you.

- **Deselectable in 2K increments.** Our deselect feature enables you to switch off any 2K to avoid overlap with your existing memory.
- **Fully socketed memory.** This feature enables you to expand the memory board yourself.
- **Plug selectable addressing.** Now you can re-address without soldering

Other Standard Features

- Power-saving dynamic board with on-board invisible refresh
- One-year guarantee on parts and labor
- S-100 and Z-80 compatible

Specifications

Storage Capacity	16K, 32K, 48K, or 64K
Addressing	16K boundaries
Max. Input Load	One LS TTL load
Output Buffering	On all data lines
Access Time	450ns
Cycle Time	480ns
Wait States Generated	None
Maximum DMA Rate	1 Mhz

Circle 45 on inquiry card.

Power Consumption

+16	150ma
+ 8	300ma
−16	20ma

Other Products From Central Data

Central Data also offers, fully-assembled and in kits, a range of other computer products including (pictured above) our 2650 microprocessor, software packages, TV/Monitor, ASCII keyboard, and (not pictured) floppy disk system, to name just a few.

For More Information
To Place an Order

We welcome any questions you have about our RAM boards or other products. To place an order, or for more information contact:

Central Data Corporation

P.O. Box 2484, Station A
Champaign, IL 61820
Ph. (217) 359-8010

Place orders prepaid or COD. Delivery is stock to 30 days with shipping and handling prepaid in Continental United States. Please include phone as well as name and address.

An Easy Programming System

Joseph Weisbecker
1220 Wayne Av
Cherry Hill NJ 08002

This article describes a hexadecimal interpretive programming system which requires less hardware than high level languages such as BASIC, and which I feel is much easier to use than machine language. In my experience, hexadecimal interpretive programming is ideally suited to real time control, video graphics, games or music synthesis. It can be used with inexpensive computer systems consisting of a hexadecimal keyboard and only 1 K or 2 K of programmable memory. Expensive terminals and large memories aren't required. You can quickly and easily write useful programs that require five to ten times less memory than conventional high level languages without resorting to the tedious complexities of actual machine language.

Interpretive Programming

This programming approach isn't new, but surprisingly few people seem to be using it. The technique consists of designing a high level pseudomachine language that is more powerful for specific applications than conventional machine language. An interpretive program is then written to execute this new set of pseudoinstructions. Each pseudoinstruction is really just a code that specifies a machine language subroutine. This set of subroutines can be designed to perform any functions you might need for your application. By staying with a machine language format, and not using labels or English words for instruction codes, memory requirements are lower. By limiting the addressing range and number of variables, you can limit each pseudoinstruction code length to several bytes for further memory space savings. Interpretive programs for these powerful pseudomachine languages can require as few as 512 bytes of memory. It has seldom taken me more than a week to implement a new hexadecimal interpretive language, and I can then use it for years. The approach can be thought of as vertical microprogramming with the microprocessor machine language used as the microcode representation.

To illustrate the compactness of these types of programs, I wrote a video tic-tac-toe program using the CHIP-8 language described below. Only 500 program bytes were required versus 3000 bytes for an equivalent version written in BASIC. Besides saving memory, this also meant 2000 fewer keystrokes for initial program entry. In addition, the CHIP-8 interpreter was about eight times smaller than the BASIC interpreter. The CHIP-8 program ran on a 1.5 K memory system with a hexadecimal keyboard, while the BASIC program required an 8 K system with an ASCII keyboard and alphanumeric display. The CHIP-8 program took about 12 hours to design, hand code, enter and debug. I suspect that the BASIC version took at least as long on a much more expensive system.

This hexadecimal interpretive programming approach is important for two reasons. First, it reduces the cost of the hardware you need to get started in home computing. Second, it drastically reduces the amount of read only memory required in microprocessor based products such as controllers and video games. Read only memory cost is a significant factor in these types of products.

A detailed example will be used to illustrate the hexadecimal intepretive programming approach. The new RCA COSMAC VIP computer will be used for this example (see August 1977 BYTE, page 30, for a description of this computer). It is a low cost, single card computer containing 2048 bytes of programmable memory, a graphic video display, and a hexadecimal keyboard. I had this type of programming in mind when I incorporated features such as multiple program counters in the COSMAC (1802) microprocessor architecture.

The pseudomachine language used in my example will be one called CHIP-8, designed for use with the COSMAC VIP system. I will

discuss using this language rather than describing the interpreter for it. Suffice it to say that the interpreter only requires 512 bytes and resides at memory locations 0000 to 01FF (hexadecimal). Programs written in the CHIP-8 language must start at memory location 0200 (hexadecimal). The sample program described will run on a 1024 byte memory system. This includes the CHIP-8 interpreter, the program, work area and video display refresh buffer. The program itself only requires 60 CHIP-8 instructions.

CHIP-8 Language

Table 1 describes the 31 CHIP-8 instructions provided in this pseudomachine language. Each instruction requires only two bytes (four hexadecimal digits). Memory addressing is limited to 4096 bytes so that only three hexadecimal digits are needed to specify a memory address. The number of variables has been limited to 16, labeled V0 to VF in this article. These are 1 byte variables or registers that can be modified or examined by CHIP-8 instructions. There is also a 2 byte memory address register called I, which is used by certain instructions. A real time clock or timer is provided. This timer can be set to any hexadecimal value between 00 and FF by the FX15 instruction. For example, if V2 contained hexadecimal 0A, an F215 instruction would set the timer to 0A. This timer is automatically decremented by one 60 times per second until it reaches 00. If the timer was set to 3C (decimal 60) it would reach 00 exactly 1 second later. This timer can be used to provide delays in game or control programs. A tone clock is also provided which can be set to cause a tone lasting from 1/60 to about 4 seconds.

An important feature of this type of language is that all variables (registers) are contained in memory. This means that debugging is generally lmited to examining memory locations, not internal microprocessor hardware registers. Astute readers will be wondering why I maintained a fixed 2 byte instruction length when variable instruction length was possible. Since absolute memory addresses are used, fixed 2 byte instructions avoid addressing confusion that increases programming errors. Also, any instruction can easily be replaced by a branch instruction of the same length for debugging breakpoints or program patching.

Graphic Display Approach

Before proceeding with a detailed programming example, readers will need to understand the video display system. Figure 1 shows the graphic display format used.

(Hexadecimal) Instruction	Operation
1MMM	Go to 0MMM
BMMM	Go to 0MMM + V0
2MMM	Do subroutine at 0MMM (must end with 00EE)
00EE	Return from subroutine
3XKK	Skip next instruction if VX = KK
4XKK	Skip next instruction if VX ≠ KK
5XY0	Skip next instruction if VX = VY
9XY0	Skip next instruction if VX ≠ VY
EX9E	Skip next instruction if VX = hexadecimal key (LSD)
EXA1	Skip next instruction if VX ≠ hexadecimal key (LSD)
6XKK	Let VX = KK
CXKK	Let VX = Random Byte (KK = Mask)
7XKK	Let VX = VX + KK
8XY0	Let VX = VY
8XY1	Let VX = VX/VY (VF changed)
8XY2	Let VX = VX & VY (VF changed)
8XY4	Let VX = VX + VY (VF = 00 if VX + VY ≤ FF, VF = 01 if VX + VY > FF)
8XY5	Let VX = VX – VY (VF = 00 if VX < VY, VF = 01 if VX ≥ VY)
FX07	Let VX = current timer value
FX0A	Let VX = hexadecimal key digit (waits for any key pressed)
FX15	Set timer = VX (01 = 1/60 second)
FX18	Set tone duration = VX (01 = 1/60 second)
AMMM	Let I = 0MMM
FX1E	Let I = I + VX
FX29	Let I = 5 byte display pattern for LSD of VX
FX33	Let MI = 3 decimal digit equivalent of VX (I unchanged)
FX55	Let MI = V0 : VX (I = I + X + 1)
FX65	Let V0 : VX = MI (I = I + X + 1)
00E0	Erase display (all 0s)
DXYN	Show n byte MI pattern at VX-VY coordinates. I unchanged. MI pattern is combined with existing display via EXCLUSIVE-OR function. VF = 01 if a 1 in MI pattern matches 1 in existing display.
0MMM	Do machine language subroutine at 0MMM (subroutine must end with D4 byte)

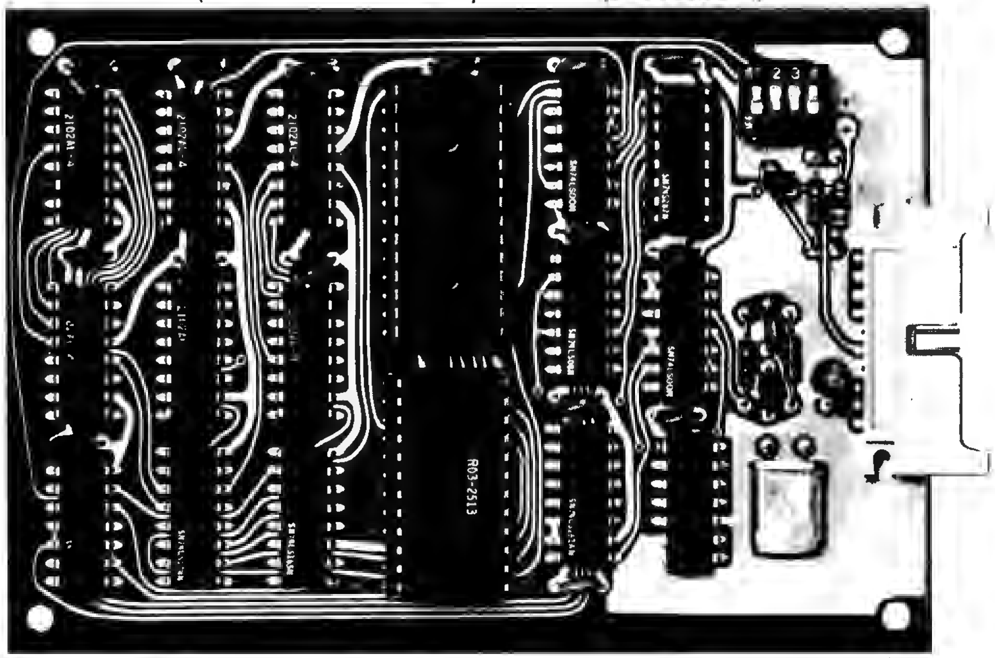
Table 1: CHIP-8 instruction set. Note that invalid hexadecimal characters in the hexadecimal instructions listed are replaced by valid hexadecimal codes when a program is written. Thus B1000 might be a valid use of the BMMM instruction.

Everything you ever wanted in a video interface.

(Automatic scrolling, cursor, erase functions, etc.)

And less.

(Stand-alone—requires no processor.)



VIB-1000. Shown actual size.

NPC's NEW VIB SERIES. FROM $129⁹⁵

Don't let the low price fool you. These are not kits. They are fully assembled and tested boards. NPC's custom CRT Processor Chip is the secret behind the entire family. All perform character entry and cursor control functions automatically. They execute all normal cursor commands as well as more sophisticated functions such as erase page, erase to end of line, etc. Text automatically scrolls when cursor reaches bottom of page. Display format is fixed at 16 lines of 64 characters for optimum efficiency.

VIB-1000 All standard features described above. $129.95

VIB-2000 All features of the VIB-1000 plus FOREGROUND AND BACKGROUND video modes, SCREEN PROTECT and READY/BUSY handshake. $159.95

VIB-3000 All features of the VIB-2000 plus SCREEN READ capability. $169.95

VIB-2000S Serial interface (RS-232C) version of the VIB-2000. $189.95

Where Pricing is as Important as Technology

NPC Nucleonic Products Company

6660 Variel Avenue
Canoga Park, California 91303 • (213) 887-1010
Telex 69-8481 • TWX 910-494-1954
A Division of DuMont Electronics Corporation

If your local computer store does not have NPC's new VIB series yet, contact the distributor nearest you to order direct.

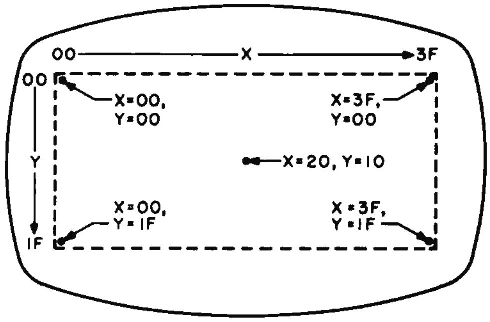
STEP SYSTEMS, INC.
4815 West 77th Avenue
Minneapolis, MN 55435
(612) 831-3644

NATIONAL ELECTRO SALES
12063 W. Jefferson Blvd.
Culver City, CA 90230
(213) 391-6294

DEALER INQUIRIES INVITED

Circle 287 on inquiry card.

Figure 1: A drawing of the video display. The inner dashed square is the playing area. The range of X and Y is shown.



The dotted line indicates the area of the screen used for display. This display area consists of an array of spot positions 64 wide by 32 high. These spot positions represent bits in a 256 byte page of memory. When a memory bit is one, the spot position is on (white). The CHIP-8 language specifies spot positions on the screen by an XY coordinate system as shown in figure 1. The values of the X coordinate (horizontal spot position) can run from 00 to 3F (0 to 63 decimal). The values of the Y coordinate (vertical spot position) run from 00 to 1F (0 to 31 decimal). Any two variables (V0 to VF) can be used to specify the X and Y coordinates of a spot position on the screen.

The display instruction (DXYN) lets you show a pattern of spots on the screen. This pattern of spots can form a picture, letter, number, etc. Patterns are represented in memory by a list of one to 15 bytes. Suppose you want to display a rocket ship. You must first construct a rocket ship pattern on grid paper as illustrated in figure 2. The hexadecimal codes for this pattern can then be derived directly from the bit pattern.

To show this rocket ship on the screen with a DXYN instruction, you must first set I to the address of the rocket ship pattern byte list in memory. You must then set two variables to the X and Y coordinates at which you want the rocket ship pattern to appear on the screen. The X and Y coordinates specify the position of bit 7 of the first pattern byte on the screen. For example, the following short program would show the rocket pattern of figure 2 at the top left corner of the screen:

Memory Address (Hexadecimal)	Instruction Code	Comments
0200	6200	Set V2 = rocket X coordinate = 00
0202	6300	Set V3 = rocket Y coordinate = 00
0204	A20A	Set I = rocket pattern address = 020A
0206	D236	Display 6 byte rocket pattern
0208	1208	End loop
020A	2070	Rocket pattern byte list
020C	70F8	
020E	D888	

The last hexadecimal digit of the display instruction (DXYN) must always specify the number of bytes in the pattern to be shown on the screen. The DXYN instruction compares each bit of the new pattern to be displayed with whatever is already displayed on the screen at the same spot positions. If a 1 bit is already displayed at the same position as a 1 bit in the new pattern to be displayed, a 0 will be shown on the screen at this spot position, and VF will be set to 01. In other words, the new pattern to be shown is combined with the pattern already showing on the screen via an EXCLUSIVE OR function. This means that after a pattern is shown on the screen it can be erased by showing the same pattern again with the same X and Y coordinates. Incrementing the X or Y coordinate and showing the pattern a third time would cause the illusion of motion. If the value of VF is 01 after showing the pattern on the screen, it means that the pattern touched or hit a previously displayed pattern.

The DXYN instruction permits displaying, erasing and moving individual patterns on the video screen. The ability to detect when one pattern meets another permits you to program chase, paddle and target games.

Decimal Digits and Random Bytes

Several instructions are provided to permit displaying decimal numbers on the video screen. These are useful for game scorekeeping, etc. An FX33 instruction converts the value of any variable (VX) to decimal form. Suppose I=0442 and V9=A7 (hexadecimal). An F933 would cause 01 to be stored in memory location 0422 (hexadecimal), 06 in 0423, and 07 in location 0424.

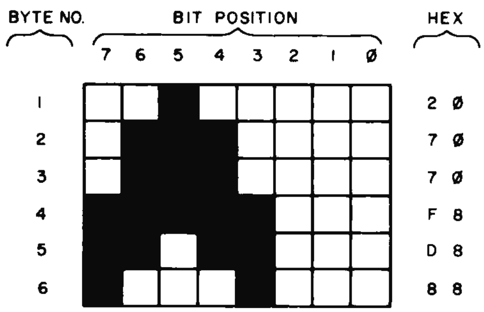


Figure 2: The definition of the rocket pattern is shown. The dark squares are encoded as a 1 bit in the appropriate byte. The actual value of each byte of the pattern is shown under the HEX column.

ARTEC CRAFTSMANSHIP HAS CREATED

The First Truly Silent Motherboard

Noise in your bus lines means errors in your programs. The Artec shielded Motherboard totally eliminates noise.

At 4MHz, the Artec shielded Motherboard is free from spurious noise. No ringing in your bus lines. No errors in your programs.

This Motherboard offers you engineering and craftsmanship never before available in the small computer field. Outstanding as either a replacement for your present Motherboard or as the heart of a new system. Consider these features:

- ⅛th inch thick—more than twice as thick as most Motherboards.
- Totally shielded—all holes plated through; full bus terminations.
- Fits easily into any standard chassis.
- Masterite edge connectors—the finest quality connectors available.
- Reflowed solder circuitry.
- No soldering required.
- Designed for the S-100 bus.

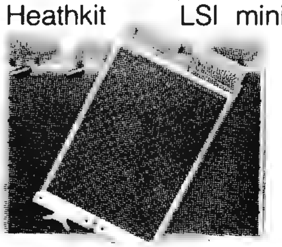
The Motherboard price is: $150(KIT)
$190 (ASSEMBLED)

Five years of experience in every card

For five years, Artec has worked hard to develop a complete line of custom, prototype and off-the-shelf printed circuit boards. And in five years of tough industrial use, Artec boards have proven themselves among the most reliable boards available anywhere.

NEW! DEC® and Heath Compatible LSI Boards

The new Artec WW11 lets you adapt or add onto your DEC LSI-11 or Heathkit LSI minicomputer. Can accommodate 14 and 16 pin DIPs plus all necessary passive components.



FULL CARD $75
(10.45" x 8.4")
HALF CARD $35
(5.225" x 8.4")

® Trademark of Digital Equipment Corporation

Order today!

Put an Artec board to work for you. Use your Mastercharge or Visa. Or just send along a money order. We can accept only U.S. currency. Please include $3 handling on all orders. California residents add 6% sales tax.

Please send me: (include quantity)

___ Shielded Motherboard   ___ Full WW11 Card   ___ Half WW11 Card

☐ I've enclosed a money order for $ __________

☐ Mastercharge ☐ Visa __________ (number) (exp. date)

Name __________

Address __________

City _____ State _____ Zip _____

Calif. Res. add 6% sales tax. Encl. $3 handling. 10% discount for students and computer club members. (Please enclose name of club or school)

ARTEC ELECTRONICS, INC.

Artec Electronics, Inc. • 605 Old County Rd. • San Carlos, CA 94070
(415) 592-2740

Circle 10 on inquiry card.

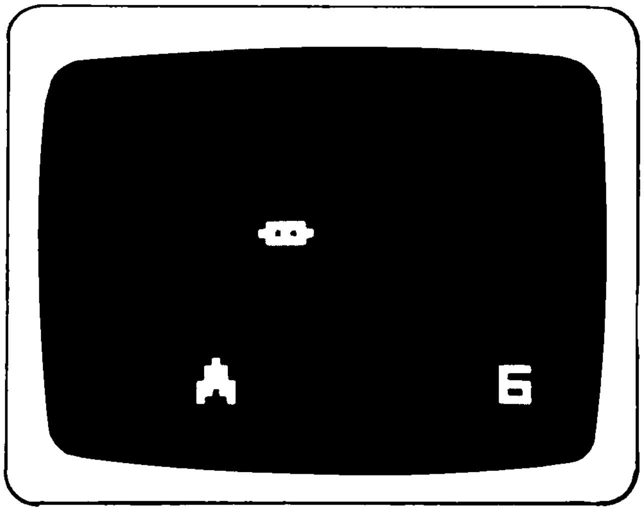


Photo 1: The actual video display of the game showing the rocket, UFO and score.

Since A7 in hexadecimal equals 167 in decimal, we see that the three bytes addressed by I represent the decimal equivalent of the value of V9. If I=0422, an F265 instruction could then be used to set V0, V1 and V2 to the values of the three bytes addressed by I above (01, 06 and 07). An FX29 instruction can then be used to set I to a 5 byte pattern representing any one of the three decimal digits. An F229 instruction would leave I addressing a 5 byte pattern for displaying the least significant decimal digit (7 in this example). A DXY5 instruction can then be used to display the decimal digit on the video screen at any desired position.

The above example illustrates the use of an FX65 instruction to transfer three memory bytes to three variables (V0 to V2). The FX55 instruction will store any number of variables in memory locations starting at the I address. These two instructions can be used to increase the number of variables by swapping sets of variables and memory bytes. Just remember that variables are always copied to or from memory in groups starting with V0 and ending with VX, inclusive.

It is often useful to generate random byte values. The CXKK instruction sets any variable (VX) to a random byte value. This random byte will have any bits matching 0 bit positions in KK (a 2 digit hexadecimal number) set to 0. For example, a C407 instruction would set V4 equal to a random byte value between hexadecimal values 00 and 07.

The remainder of the CHIP-8 instructions should be self-explanatory. The 2MMM instruction will transfer control to a subroutine which must be terminated by

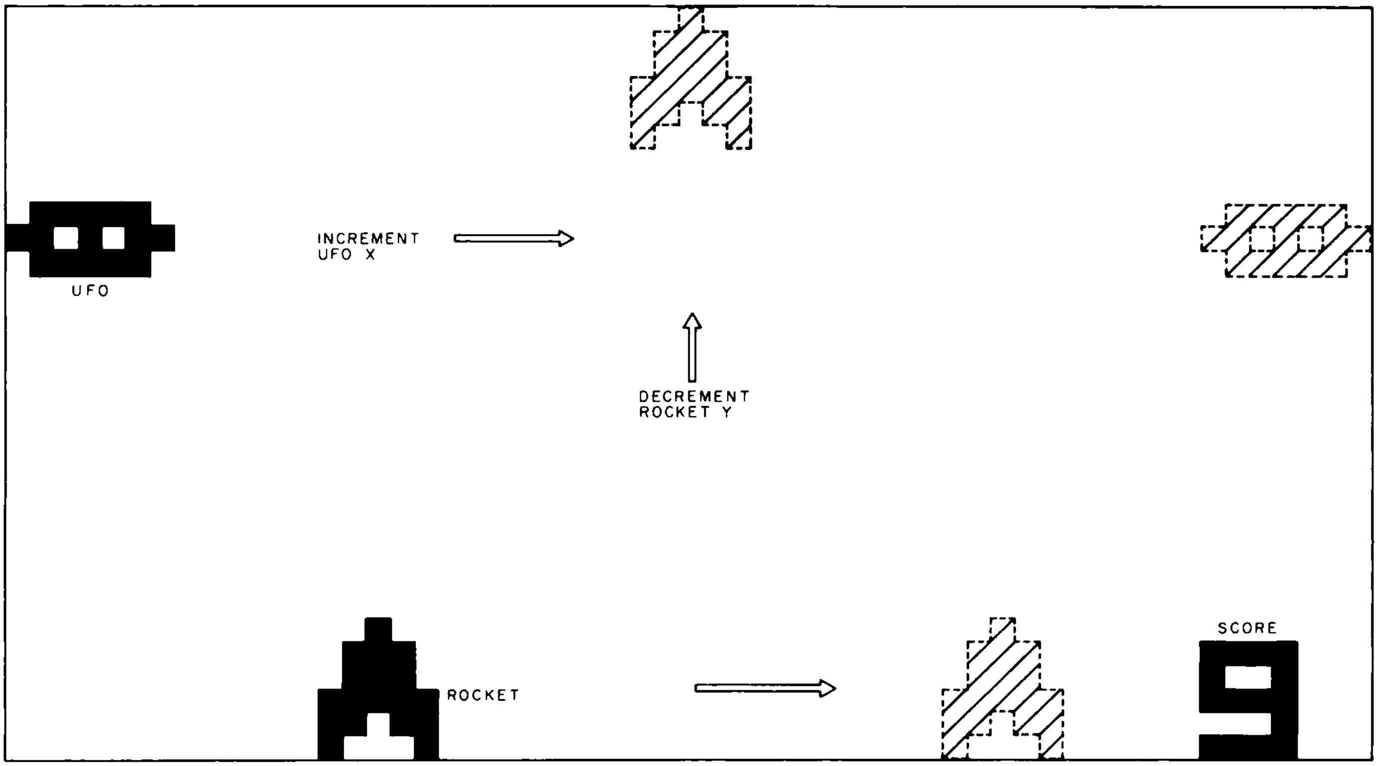
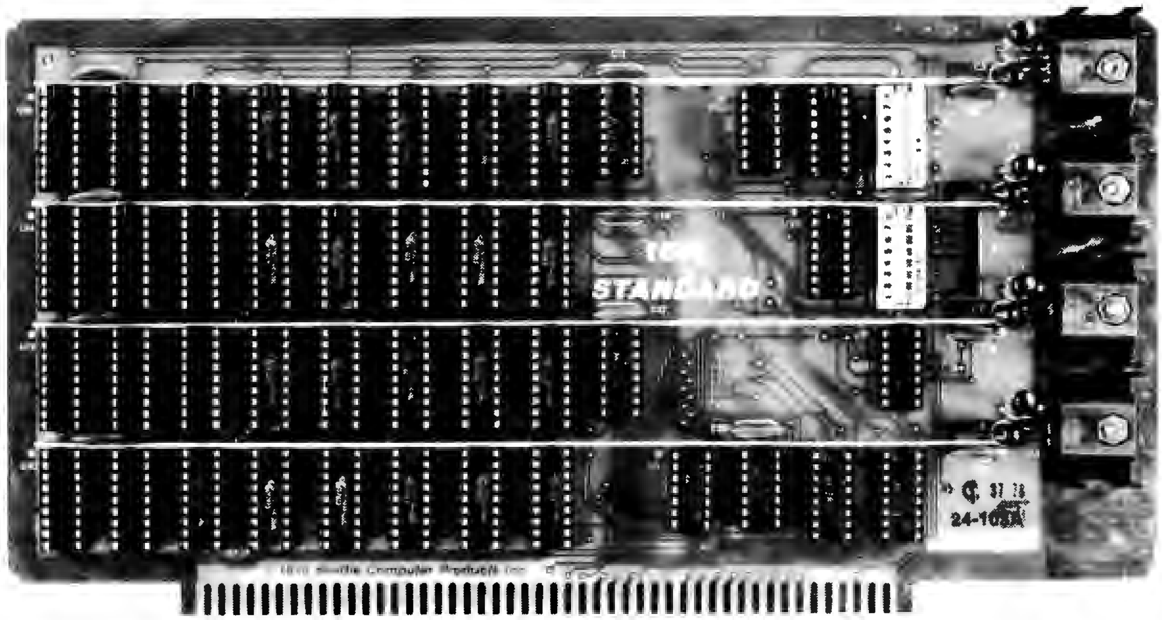


Figure 3: The range of rocket X values is from hexadecimal 0F to 2E. Rocket Y is decremented from hexadecimal 1A to 00. The UFO Y remains a constant hexadecimal 08, while the UFO X is incremented from hexadecimal 00 to 39.



16K STANDARD RAM— $285

KIT — 450 nsec. Ideal for SOL 20, IMSAI 8080, etc.

For 4 Mhz. systems—order our board with 250 nsec. memory chips—priced at $320. For either board—full factory assembly, burn-in, testing, 1-year warranty on both parts and labor—add $25.

- ***DIP Switch Addressing***
- ***First Quality Components***
- ***All Inputs Buffered***
- ***Off-the-Shelf Availability***
- ***Fully Static TMS 4044***
- ***Full Year Guarantee***

Circle our Reader Service Card Number for free Catalog

Contains a complete description of our products including the recently announced 16K PLUS premium RAM which is ideal for Cromemco systems (see October ads). Also contains parts, 8K RAM versions, schematics, specifications, full warranty statements, previews of new products such as our 8086 CPU board.

Texas Instruments

TMS 4044 – $5

450 nsec. — Quantity 64

Factory Prime memory chips, guaranteed good.

	250 nsec.	450 nsec.
1-31 chips	**$8**	**$7**
32-63 chips	7	6
64-over	6	5

For orders under $50 — add $2 handling.

Guarantee: Kits — All parts guaranteed for one year. Factory assembled units for USA customers — Both parts and labor guaranteed for one year; ten day return privilege. Factory assembled units for foreign customers — parts guaranteed for one year; no return privilege.

Direct Factory Orders — you may phone for VISA, MC, COD orders. ($3 handling charge for COD orders only). Purchase orders accepted from recognized institutions. Personal checks OK but must clear before shipment. Shipped prepaid with cross-country orders sent by air. Shipping — normally within 48 hours. Washington residents add 5.4% tax. Our 16K STANDARD and Premium 16K PLUS board are also available through most computer stores.

Seattle Computer Products, Inc.
1114 Industry Drive, Seattle, WA. 98188
(206) 255-0750

Circle 318 on inquiry card.

instruction 00EE to return control to the instruction following the 2MMM. You can nest these subroutines. The 0MMM instruction permits a machine language subroutine to be inserted if required.

Designing a Video Game Program

A detailed example will illustrate how easily the CHIP-8 language can be used to program a real time video game. The first step always involves specifying the video display and the functions to be programmed. Figure 3 shows the display format chosen for this game. An enemy UFO will be constantly moving from left to right across the top of the screen. A single digit score will be displayed at the lower right. A rocket ship will appear at a random horizontal position along the bottom edge of the display area. You can launch this rocket by pressing key F on the hexadecimal keyboard. The rocket will then move vertically toward the top of the screen. When it reaches the top or hits the target UFO it will be erased and a new rocket will appear at the bottom of the screen. After nine rockets have been launched the game ends and no new rockets will appear. If you hit the UFO with a rocket the score is incremented by 1.

After specifying the positions of the various game patterns on the video screen as shown in figure 3, you must decide on how the 16 variables will be used in the program. Table 2 illustrates how we will use the variables in this example. Six variables (V3, V4, V5, V6, V7, V8) are needed to specify the X and Y coordinates of the three types of patterns involved (score, target UFO and rocket). We need two more variables (V1, V2) to keep track of the current score and number of rockets launched. V9 will be used as a flag that shows whether or not the current rocket has been launched. VA will be set to 01 if the rocket hits the UFO (ie: point scored) and V0 will be used for a working register in the program. VF is the hit flag and is automatically set to 01 when a hit occurs.

Flowcharting the Game

I believe you should always construct a detailed flowchart, such as the one shown in figure 4. Proper flowcharting will save hours of debugging and will simplify making future changes to your program. A flowchart also lets you see the major and minor loops in your program, allowing you to avoid timing bugs that can occur in real time situations such as video games.

Step 1 involves initializing the score and rocket counters, as well as the X and Y coordinates for the target UFO and on screen score digit. The UFO pattern is shown on the screen so that it can subsequently be moved. In step 2 the latest score is shown on the screen, and V2 is checked to see if the game should end because nine rockets have been fired.

Step 4 performs the operations required to show a new rocket at the bottom of the screen. The rocket count is incremented by 1 for each new rocket. The rocket pattern Y coordinate is set to hexadecimal 1A so that the rocket will appear at the bottom of the screen. The rocket X coordinate is set to a random value between hexadecimal 0F and 2E so that it will appear at a random horizontal position without interfering with the score digit. The flag V9 is set to 00 to indicate that the rocket has not yet been fired. The rocket is then shown on the screen and the program proceeds to the loop containing steps 5, 6 and 7.

This loop causes the target UFO to continuously move across the top of the screen while waiting for key F to be pressed. The UFO is randomly moved zero, one, two or three spot positions to the right each time the loop is executed. This gives it a rather fast, randomly varying rate of motion, making it harder to hit. The movement of the UFO merely involves incrementing its X coordinate (V5). When V5 is incremented past the right edge of the display area, wrap around automatically occurs so that the UFO reappears at the left edge of the display area. This wrap around automatically occurs when any X or Y coordinate of any display pattern is incremented or decremented past any edge of the 64 by 32 bit display area.

When key F is pressed to launch the rocket, step 6 causes V9 to be set to 01. Step 7 then causes step 8 to be included in the loop. Step 8 checks the value of the system real time clock (or timer) to see if it has reached 00 yet. When the timer reaches 00 the rocket is moved up one spot position, and the timer is reset to a value of 03 (1/20

V0—	Temporary variable
V1—	Score (00 at start)
V2—	Rocket counter (00 at start)
V3—	Score X (38)
V4—	Score Y (1B)
V5—	UFO X (00 at start)
V6—	UFO Y (08)
V7—	Rocket X (random, 0F to 2E)
V8—	Rocket Y (1A at start)
V9—	Rocket fired flag (00=no, 01=yes)
VA—	Score increment (00 or 01)
VF—	Hit flag (00 or 01)

Table 2: Rocket program variables. VB, VC, VD and VE are not used in this program.

TARBELL SOFTWARE
Extensive & Inexpensive.

TARBELL CASSETTE or DISK BASIC
only $36.00

Most features of ALTAIR* Extended BASIC are included *PLUS* these added features:

- Assignment of I/O
- Alphanumeric line labels are allowed
- Read and write string data
- Unlimited length of variable names and strings
- Procedures with independent variables
- Number system 10 digits BCD integer or floating point

Included are commands unique to Tarbell BASIC which provide capabilities to:

• Chain to another program • Assign a physical device to a logical device • Drop an I/O channel previously assigned • Cause programs to be appended onto programs already in memory • Call a procedure and pass variables on the list • Cause interpreter to enter edit mode using 15 single character edit commands.

Tarbell BASIC occupies 21K of RAM.
Tarbell BASIC on cassette - $36
Tarbell BASIC on CP/M Disk - $36
Tarbell BASIC Source on paper - $25
Tarbell BASIC Source on CP/M Disk - $25

CP/M Disk Operating System

The I/O section of this software has been modified to operate with the TARBELL Floppy Disk Interface in 24K bytes of memory. Five commands permit listing of directory, typing contents of an ASCII file, renaming a file, erasing a file from disk, and saving memory on disk. Fourteen programs are included which are invoked like commands. Six source files are included for transferring between TARBELL Cassette and disk, cold-start loading, Basic I/O system with drivers, and reformatting crashed diskettes. Documentation includes a listing of BIOS and instructions to patch CBIOS for your system. Price is $100 on CP/M diskette with documentation. (CP/M is a product of Digital Research).

CP/M 1.4 Update Package

A TARBELL Update Package for those now using CP/M 1.3 is now available on diskette. The Update Package adds new commands and the ability to access four disk drives, as well as 2 new CP/M manuals, TARBELL CP/M User's Guide and a new BIOS listing. Price: $50.00.

SPOOLER

This 8080 program will save many hours of computing time. It intercepts all output to the list device, spools the output to a high-speed disk file, and directs the spooled data to a low-speed printer during unused cycle time while the CPU waits for transfer of data to and from the console. System throughput is greatly increased with the aid of SPOOLER. Output is never lost due to insufficient memory allocation. Fully compatible with the CP/M file system, SPOOLER permits parallel processing without hardware interrupt, and with minimal impact on other processes. Price: $70.00 (Copyright KLH Systems.)

BASIC-E Compiler

Designed to work with CP/M Disk Operating System this software requires a total of 20K bytes of memory. Included are 26 compiler error messages and 23 run-time error messages. Disk files may be read, written or updated by using both sequential and random access. Included are blocked and unblocked files. Price for compiler and run-time monitor on diskette is $10.00. Manual is available separately for $5.00. (Public domain software by Gordon E. Eubanks, Jr.).

CBASIC Programming System

Upward compatible from BASIC-E, CBASIC is similar but expanded to include several business oriented facilities, allowing decimal computations to 14 digits of precision, data formatting and PRINT USING statements. Statements allow access to disk files and disk file maintenance. Strings of characters may be read from the console to permit correct input line format to be checked before reading data. General programming features include variable names up to 31 characters, optional line numbers, dynamic debugging tracers, and optional data output to printer. CBASIC on diskette and manual priced at $100. (Copyright Software Systems.)

EMPL-an 8080 APL

Especially suited to educational applications, EMPL is an adaptation of APL, using the ASCII character set. Only one-dimension arrays are allowed. This 8K version occupies the first 5376 bytes of memory and operates in two modes. The Execution Mode permits all instructions to be executed immediately. The Definition Mode permits the user to enter functions. EMPL on Tarbell Cassette with manual is $15. (Copyright 1977 Erik Mueller).

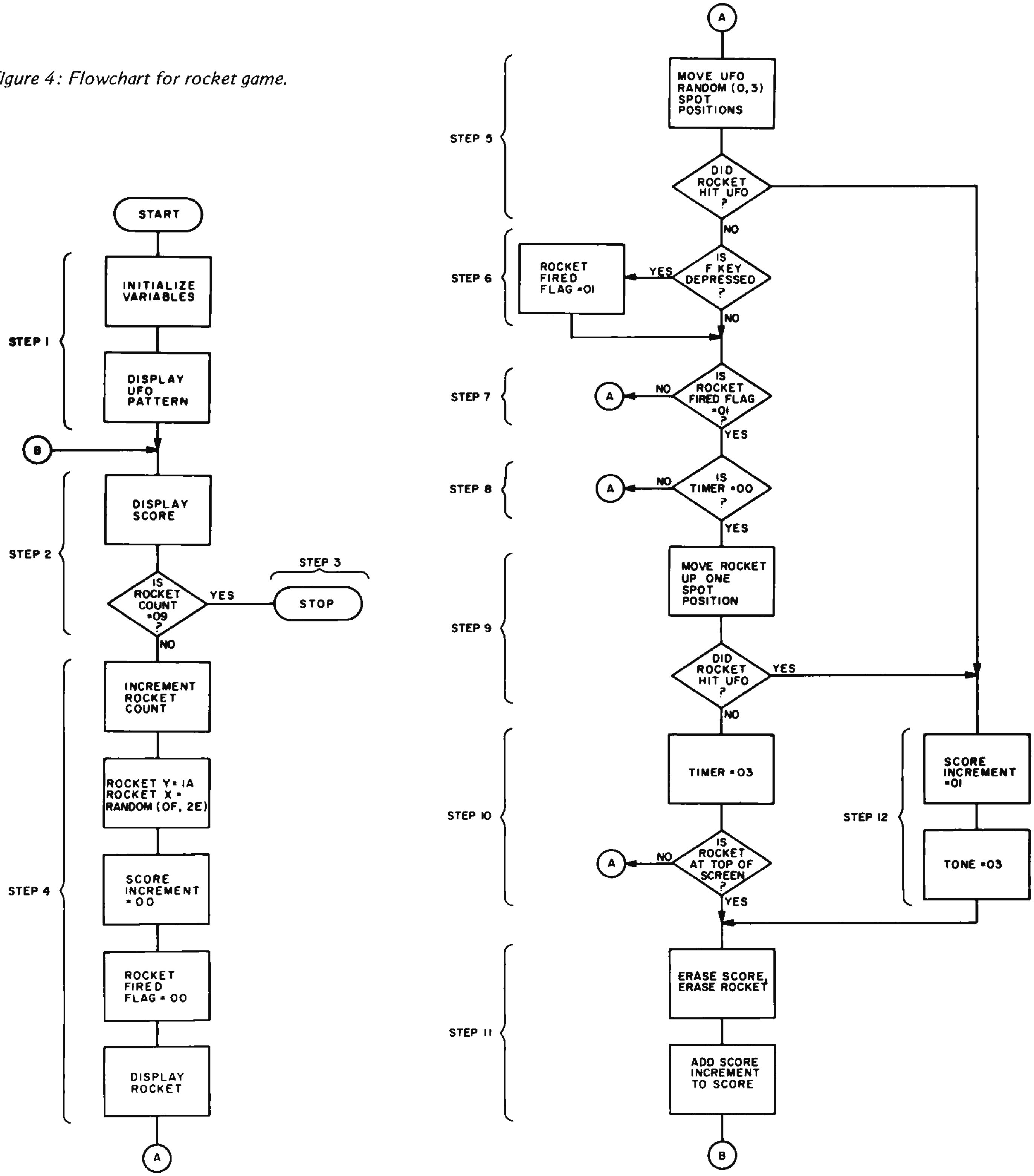
**ALTAIR is a trademark/tradename of Pertec Computer Corp.*
***CP/M is a trademark/tradename of Digital Research*

Tarbell Electronics

950 DOVLEN PLACE • SUITE B • CARSON, CALIFORNIA 90746
(213) 538-4251 • (213) 538-2254

Circle 360 on inquiry card.

Figure 4: Flowchart for rocket game.



second). This timer reset value determines the speed at which the rocket moves upward. The larger the timer value, the slower the rocket moves. The loop comprising steps 5 thru 10 keeps both the target UFO and rocket moving on the screen until either they touch each other (a hit), or the rocket reaches the top of the screen without touching the UFO.

If the rocket and UFO touch each other, step 12 is executed. This sets VA to 01 and sounds a short tone so the player knows (s)he scored. Step 11 is then executed to erase the rocket and the current score on screen. Step 11 also increments the score variable (V1) by the 01 in VA. The program then returns to step 2 where the updated score is shown on the screen. If less than nine rockets have been fired, a new rocket is shown on the screen in step 4, and the step 5-6-7 loop is repeated as before.

If the rocket reaches the top of the screen without touching the UFO, step 10 will branch to step 11 causing the score and rocket to be erased. In this case VA will contain 00 (from step 4) so that the score will remain unchanged. Note that the entire program has now been designed. By double-checking the program in flowchart form, you can eliminate almost all program bugs before they occur. An extra hour spent on the flowchart can eliminate many hours of debugging later. All that remains now is to translate the flowchart into an appropriate sequence of CHIP-8 instructions.

Address (Hexadecimal)	Instruction	Pseudocode	Comments	
0200	6100	V1=00	Step 1:	Score
0202	6200	V2=00		Rocket count
0204	6338	V3=38		Score X
0206	641B	V4=1B		Score Y
0208	6500	V5=00		UFO X
020A	6608	V6=08		UFO Y
020C	A27E	I=027E		UFO pattern
020E	D563	SHOW 3MI@V5V6		UFO
0210	226A	DO 026A	Step 2:	Show score
0212	4209	SKIP; V2 NE 09		
0214	1214	GO 0214	Step 3:	End loop
0216	7201	V2+01	Step 4:	
0218	681A	V8=1A		Rocket Y
021A	6A00	VA=00		
021C	C71F	V7=RND		
021E	770F	V7+0F		Rocket X
0220	6900	V9=00		
0222	A278	I=0278		Rocket pattern
0224	D786	SHOW 6MI@V7V8		
0226	A27E	I=027E	Step 5:	UFO pattern
0228	D563	SHOW 3MI@V5V6		Erase UFO
022A	C003	V0=RND		
022C	8504	V5=V5+V0		Set VF
022E	D563	SHOW 3MI@V5V6		
0230	3F00	SKIP; VF EQ 00		
0232	1262	GO 0262		Step 12 if hit
0234	600F	V0=0F	Step 6:	
0236	E0A1	SKIP; V0 NE KEY		
0238	6901	V9=01		
023A	3901	SKIP; V9 EQ 01	Step 7:	
023C	1226	GO 0226		Step 5
023E	F007	V0=TIME	Step 8:	
0240	3000	SKIP; V0 EQ 00		
0242	1226	GO 0226		Step 5
0244	A278	I=0278	Step 9:	Rocket pattern
0246	D786	SHOW 6MI@V7V8		Erase rocket
0248	78FF	V8+FF		
024A	D786	SHOW 6MI@V7V8		
024C	3F00	SKIP; VF EQ 00		
024E	1262	GO 0262		Step 12
0250	6003	V0=03	Step 10:	
0252	F015	TIME=V0		
0254	3800	SKIP; V8 EQ 00		
0256	1226	GO 0226		Step 12
0258	226A	DO 026A	Step 11:	Erase score
025A	A278	I=0278		Rocket pattern
025C	D786	SHOW 6MI@V7V8		Erase rocket
025E	81A4	V1=V1+VA		Score+VA
0260	1210	GO 0210		Step 2
0262	6A01	VA=01	Step 12:	
0264	6003	V0=03		
0266	F018	TONE=V0		
0268	1258	GO 0258		Step 11
026A	A2A0	I=02A0	SSS:	3 byte work area
026C	F133	MI=V1 (3DD)		
026E	A2A2	I=02A2		Least significant digit
0270	F065	V0: V0=MI		
0272	F029	I=V0 (LSDP)		
0274	D345	SHOW 5MI@V3V4		
0276	00EE	RET		
0278	2070		ROCK:	
027A	70F8			Rocket pattern
027C	D888			
027E	7CD6		UFO:	UFO Pattern
0280	7C00			

Figure 5: The rocket program code in CHIP-8 hexadecimal interpretive language instructions. The steps specified relate directly to the flowchart given for the game in figure 4.

Coding and Debugging the Final Program

The final program is shown in figure 5. To translate the flowchart into CHIP-8 instructions, start by listing even numbered hexadecimal memory addresses in the first column, as shown in figure 5. Fill in the third column with an abbreviated description of the function to be performed by each instruction. It is usually most convenient to locate subroutines and pattern byte lists at the end of the program. Labeling the appropriate program addresses with the flowchart step numbers will also prove helpful. The actual hexadecimal codes for the CHIP-8 instructions can then be written in column 2 and entered into the COSMAC VIP memory using the hexadecimal keyboard.

To debug the program, replace the 4209 instruction at memory location 0212 with a 1212 branch instruction. When the program runs, it will stop at location 0212 since the 1212 branch loops on itself. If the UFO and a 0 digit initial score show on the video screen, you know execution was proper up to location 0212. Replace the

"My Structured Systems business software has paid for itself in labor hours saved alone."

Mr. Ken Tunnah, Colloid-A-Tron Inc., Buffalo, New York

Ken Tunnah is one of many innovators bringing the micro revolution to the small business. As a programmer, he knows computers and their languages. As a businessman, he knows business and its languages. And when Mr. Tunnah decided to microcomputerize the accounting function at Colloid-A-Tron, he turned to Structured Systems software.

Says Mr. Tunnah: ***"The program is designed from a CPA standpoint, for multiple corporations, which we have. It is flexible and gives me the ability to change reporting by profit centers easily. It is up and running quickly, and it just keeps on running. I think it's the best business software available."***

The best software available. That's what Structured Systems Group set out to create.

Structured Systems offers three sophisticated accounting systems. Our General Ledger software is big enough for multi-client write-up by the CPA, or multi-corporate reporting for the business, but small enough for the micro budget. The very comprehensive Accounts Receivable and Accounts Payable packages will operate independently, or they will coordinate with the General Ledger.

Our systems record transactions easily and correctly, and provide an audit trail from source document to financial statements. And they will maintain monthly and year-to-date information in dollars and in percentages. And they are reliable.

The three systems interact with the user to set up parameters such as format and headings, account titles and numbering, automatic billing or reminder notices, credit limits, sales reports, a check register, and much more.

The software is designed to run on an 8080 or Z-80 CPU with 48K of memory, dual disks with CP/M®, printer, keyboard, and CRT. To make it all work for you, we have provided the most extensive documentation and support in the industry.

We provide the capability to computerize complex accounting functions on relatively inexpensive microcomputer equipment. Ken Tunnah has told us what that means: ***"I've bucked some trends. I looked around, and decided that with the right software, I could get a micro to outperform a $45,000 mini. I'm satisfied. It's simple economics."***

We can refer you to a growing number of sophisticated retailers experienced in Structured Systems Group business systems. Or we can work directly with you. We'd be happy to provide you with more information on our product line, which includes QSORT™ (a sort/merge program), CBASIC (a business BASIC), NAD™ (a mailing and addressing system), and our General Ledger, Accounts Receivable, and Accounts Payable packages.

*CP/M is a registered trademark of Digital Research

Structured Systems Group
INCORPORATED
5208 Claremont Avenue
Oakland, CA 94618
Call us at (415) 547-1567. Or write Dept. B7.

Putting the Microcomputer in Business.

1212 branch with the original 4209 instruction and put a similar idle loop branch further down in the program for the next test run. In this way you can identify which program steps are causing a problem. If you need to change any portion of the program, just insert a branch instruction to a patch added at the end. Designing, coding and debugging this simple game program required about eight hours. Actual coding and loading the program into memory required less than an hour of this time.

The sample program was kept simple for ease of understanding. Even in this simplified form it is a challenging game to play. The speeds of the rocket and UFO can be easily adjusted to make scoring more or less difficult. Adding multiple targets and 2 digit scoring is possible. Multiple rocket launch angles or after-launch steering could be incorporated. Exploding UFO patterns could be shown when one is hit.

Conclusions

Hexadecimal interpretive programming provides an easy way to program small computers. This approach requires fewer instructions and is much easier than machine language programming. On the other hand, hexadecimal interpretive programming requires much less hardware overhead cost than do high level languages such as BASIC.

A detailed example was provided to illustrate this interpretive approach for a real time video game. The RCA COSMAC VIP and CHIP-8 language were used in this example, although other hexadecimal interpretive languages are possible, and a similar approach can be used with other microcomputers. The steps required in programming with a language such as CHIP-8 are the same as required when using any language: you must specify the functions required, decide on variable and memory utilization, prepare a flowchart, check the flowchart, do the detailed coding, load the code, and debug to the extent required to get the program running properly. Only the last two steps involve using the hardware. Skipping any of the earlier steps will invariably lead to excessive machine debugging time no matter what language is used.

If you've never tried a language such as CHIP-8, you may be surprised at how easy it is to use. If you have a limited budget you will certainly appreciate the savings in hardware over conventional high level languages. Last but not least, you might even discover that designing your own hexadecimal interpretive language is also fun.■

WE'VE JUST BUILT YOU A BETTER BUG TRAP

6800 Debug Package

The TSC 6800 Debug Package provides a better way to trap program bugs. It is an extremely powerful and complete assembler language program debugging tool which is capable of simulating all functions of the 6800 microprocessor, including interrupts and I/O operations. It is an ideal substitute for hardware logic analyzers or CPU emulators at only a fraction of the cost.

Any number of breakpoints may be user defined. Each breakpoint may invoke any one or combination of eight different actions. These actions may be dependent on a user defined condition such as register A=$FF or memory location $1B55=0. The actions may also be delayed or limited by a pass count. Histogram breakpoints may be set to enable profiling of the executed program. Breakpoints may be set in RAM or ROM!

Complete simulation control allows trace mode to be enabled at anytime. During trace, registers and opcode mnemonics are displayed after each instruction is executed. Single or multiple instruction stepping is permitted as well as simulation speed control. The trace back feature allows the past 256 executed instructions to be viewed. Program execution may be halted at anytime by operator command.

Memory protection and traps are another key feature. Any section(s) of memory may be write, execute, memory, or simulate protected. Execution traps allow program exit on general conditions such as interrupt instruction, transfer instruction, subroutine nest count, and instruction count timeout.

General features include a line at a time assembler, disassembler, memory interrogation commands, hex calculator, machine states counter, stack protection, register modifier, and mode control. In all, there are over 50 commands available. The manual includes detailed operating instructions as well as the complete commented source listing. Requires 9K at $3C00.

SL68-30	**Manual and source listing**	**$35.00**
SL68-30C	**with KCS Cassette**	**$41.95**
SL68-30D	**with mini FLEX™ diskette**	**$43.00**
SL68-30F	**with 8" FLEX™ diskette**	**$55.00**

Send 25¢ for a complete catalog of TSC's assembler language software for the 6800, 8080, and 6502.

TSC™ Technical Systems Consultants, Inc.
Box 2574 W. Lafayette, IN 47906

Specialists in Software & Hardware for Industry & the Hobbyist

Circle 370 on inquiry card.

Prof George A Gerhold
Dept of Chemistry
Western Washington University
Bellingham WA 98225

Teaching With a Microcomputer

Who doesn't want a tutor who is infinitely patient, expert on almost any subject under the sun, available at your beck and call, adapted to your learning speed and style, and cheap? This has attracted the attention of a number of the manufacturers of personal computers, and several of them make prominent reference to educational applications in their advertising. But the customer who uses educational applications to justify raiding the family budget for a computer will have some explaining to do, for good teaching software on microcomputers is not available, nor is it easy to write. Why?

If you examine the 20 years of development of computer assisted instruction (CAI) on mainframe computers, you'll see that the computer can be an effective teaching tool when used properly. Students taught with a computer perform as well or better than comparison groups; they may learn two or three times faster. Their failure rate is lower and they express satisfaction with the technique. Why then is the technique not more widely used? There are three main reasons.

First, teachers are conservative. In fairness we also must realize that teachers have heard too many extravagant claims for the miracle that this or that piece of educational technology will produce; their cynicism is understandable. Proponents of computer assisted instruction have not always been conservative (nor have they always been accurate). The cost of hardware is also seen to be prohibitive. Exceptions exist, but the most visible (ie: highly funded) systems are costly to purchase or lease. Third, the creation of high quality software is a difficult and time consuming task which to this point has provided little compensation beyond personal satisfaction.

The personal computer will eliminate two of the reasons. Teachers are discovering computers through their own efforts and through stimulation provided by students and parents who have computers in their homes. The major price breaks in the cost of hardware have resulted from the introduction of smaller machines. Minicomputer systems cost less than 10 percent of mainframe based systems, and microcomputer systems will possibly cost less than 10 percent of minicomputer systems (using in all cases initial capital outlay, the most relevant number for individuals or small institutions). What remains is to create appropriate software for teaching with a microcomputer.

Before we attempt to write good teaching software, there are some rather fundamental questions to be answered. First, what do we mean by good teaching? A more useful question might be, "What is good teaching not?" One relevant answer for computer teaching is, "Good teaching is not just testing."

Most programs described as teaching programs are programmed tests; the format is exclusively question, accept answer, one message for right, and a second for wrong. Random selection of messages from a list may defer boredom a bit, but that feature alone does not change a test into teaching.

Good teaching is not repetitious to the point of boredom. That's an obvious statement, but it poses a dilemma for those who would teach with the computer, because the efficient use of a computer usually involves repeated use of sections of code. The resolution of the dilemma is to write long and varied course software which can be used by a large number of students.

Good teaching does not force each student to proceed by the same path. Addition of hints or partial solutions for every question on a programmed test does not make that program a good teaching program. If each wrong answer is diagnosed, and a hint or partial solution which builds on the correct portions of that particular wrong answer is given, then we may have a good teaching tool.

Good teaching does not consist of a random collection of available bits and pieces. This implies that we should think in terms of sizable units which can become significant components of a course or subject.

Finally, we should not forget that good teachers are most often experienced teachers and that any occasional lack of enthusiasm on their part about the teaching efforts of well-meaning parents is not invariably misguided.

Another important question is, "What are we trying to teach?" This question is especially important for the personal computer user because the output devices commonly used cannot provide the notation which the students use elsewhere. Books are not written in uppercase only; exponents are not usually written with arrows or double asterisks; yes and 1 are not synonymous, nor are no and 0; the answer to every question is not a, or b, or c, or d — none of the above. If we attempt to teach using devices which impose notation limitations, we ask ourselves repeatedly, "Are we teaching what we want to teach, or are we teaching how to use and cope with the limitations of the software?"

With these points in mind let us now consider computer languages for teaching. Some teaching languages are based on a teaching strategy; others are based on software functions. Our experience is that the latter types are far superior to the former, for they allow implementation of a variety of teaching strategies. Examination of a large variety of good computer assisted instruction materials shows that they are built from a small number of operations.

For example, one must be able to send text to and accept text from the terminal. Call these functions type and accept (or T and A). (The notation herein is the PILOT notation; for a more complete description of the language see "Computer Assisted Instruction on a Microcomputer," November 1978 BYTE, page 90.)

Having accepted text, one must be able to analyze it. This is usually done with some type of a match (M) algorithm. One also needs some kind of jump (J) instruction, instructions for subroutine calls and returns (U and E), and some kind of compute (C) instruction so that one can use the full range of numerical and string operations normally associated with computers. Finally one needs some way to make execution of at least some of these instructions (at a minimum, the jump) dependent on the values of various variables or on the success or failure of certain matches.

The obvious question for the microcomputer fan is, "Can I use BASIC?" Unfortunately, the answer is, "Only with extreme difficulty." Typing text is no problem, and accepting input from the terminal can be handled. Accepting an input of *two* when you programmed INPUT X and expected the answer *2* will take some extra code, but we've already learned that these are going to be long programs by usual computer standards. The difficulties with accepting data pale in significance when compared to the difficulties with match.

Consider an extremely simple case: a question that can be answered yes. Write BASIC code which will match any of the following:

yes, Yes, YES, O.K., OK, Of course, Sure, Always.

but which will not match any of these:

yesterday, yes and no, yes or no, Alyeska, eyes.

In a good teaching language it can be done in a single line. It could be done in BASIC, at least in a BASIC with a full range of string operators, but in practice no one bothers because it's so much easier to tell the student to answer 1 for yes. One could program the match algorithm of a good teaching language in BASIC as a subroutine, but the resulting code is too slow. It seems then that BASIC (and other computational languages, such as FORTRAN and APL) are not suitable for this purpose.

Fortunately one of the best teaching languages, PILOT, is well suited for microcomputers. The original form of PILOT is too limited for production of top quality teaching materials. Several extended forms of PILOT have been developed, and the National Library of Medicine is supporting an effort to achieve a national standard for the extended language. At Western Washington University we have implemented what promises to be essentially this standard on a SwTPC 6800, and we are currently working on 8080, Z-80 and Pascal implementations. By doing so we have shown that it is possible to implement a language that includes all the operations necessary for teaching, including the full range of numeric and string operators, full floating point, and numeric and string arrays, all in a 20 K byte microcomputer. Moreover, the response time is excellent. We maintain that there is no reason to settle for less in an instructional language.

Now that you're convinced that you'll have to get better systems software, what about hardware? It appears that any of the standard microcomputers will be suitable for this application if they can accept sufficient memory (16 K to 20 K bytes). The length of instructional programs and the distance and complexity of branching within the program requires the use of floppy disks or other forms of mass storage.

A typical intructional dialogue program occupies about 8 K bytes for every 5 minutes of instruction. Any individual student might leapfrog an entire section in a few seconds if the program were written to move with a well-prepared student. Thus the system must be interpretative, with the programs stored on disks.

Good terminals for teaching should have good graphics capabilities. At this time such terminals are too expensive ($4000 to $6700), several times the cost of the rest of the system. What is needed is a video terminal with at least a 256 by 256 dot matrix that can be superimposed on a 24 line by 80 character display (upper and lower case). Until such a unit is available, we must make do with less. However, a 40 character line is rather short for this purpose.

Finally, after all these cautions and discouragements, what can or should the owner of a microcomputer who wants to use the thing to teach do? One possibility is to search for or create games which provide practice in topics which your children have already learned in school. An obvious example is a version of Spacewar that demands fractional inputs as an exercise in fractions. A second possibility is the purchase of suitable systems software and course material. Such material is beginning to come onto the market. The third and most exciting possibility is to become involved in creating teaching material. Get suitable systems software and find an interested teacher. The teacher provides the material and the approach, and you provide the programming. If you take this route, remember that the teacher knows how to teach, that students are more varied than your imagination, and that good materials require testing and editing and retesting and reediting.■

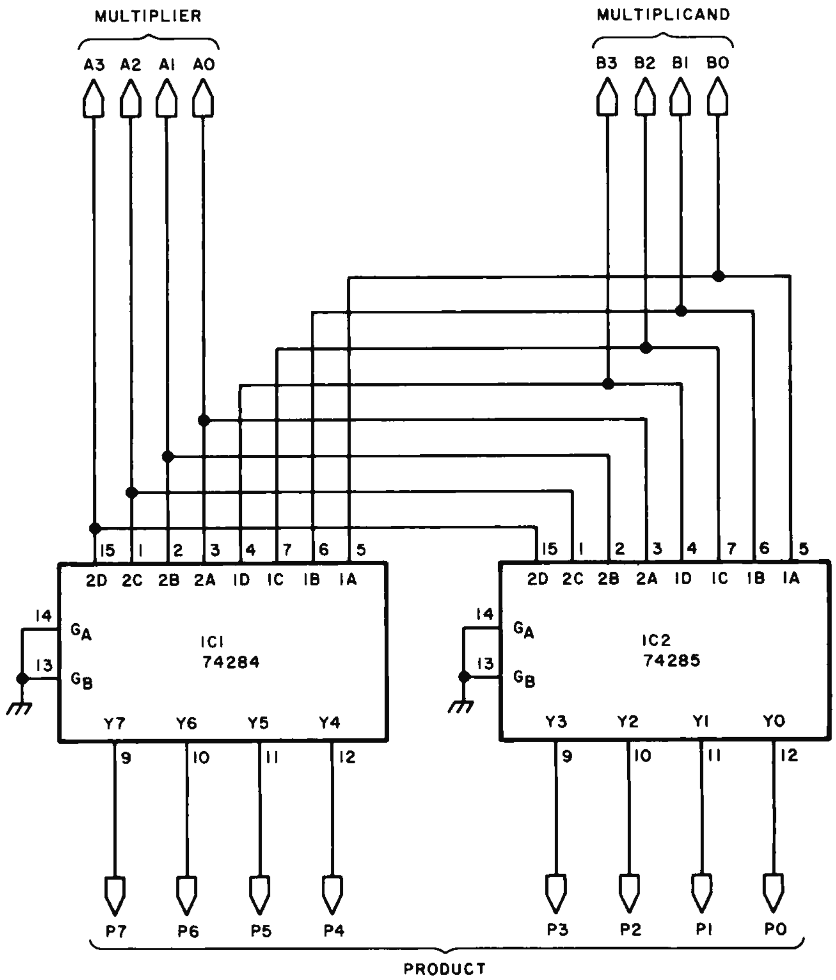
Mike Weed
Rt 2 Box 97A
Spring Valley MN 55975

Digital integrated circuits that multiply binary numbers without the use of clock pulses have been available for several years. One such 4 bit by 4 bit multiplier is illustrated in figure 1. The 8 bit product appears on the output lines about 40 ns (the propagation delay) after the input lines are set. Larger numbers may be multiplied by first sectioning the inputs into 4 bit words, forming the products of each word of one input with every word of the other input, and summing these products in the appropriate manner. The propagation delay increases approximately linearly with the number of input bits. The number of integrated circuits required increases roughly as the number of bits in the multiplier times the number of bits in the multiplicand.

Multipliers are readily constructed from simpler integrated circuits. A 4 by 4 multiplier is illustrated in figure 2. The four bits of the multiplicand are gated into the adder M places from the right if the Mth bit of the multiplier is 1. If this bit is 0, only 0 is added in. The parts count is minimized by bringing the carry output of the last adder back to a previous adder input. This system costs less than the circuit in figure 1. Its disadvantages are a higher components count and a 60 percent increase in power requirements. The two systems have comparable propagation delays.

Clockless Multiplication and Division Circuits

Figure 1: Commercially available 4 by 4 multiplier.



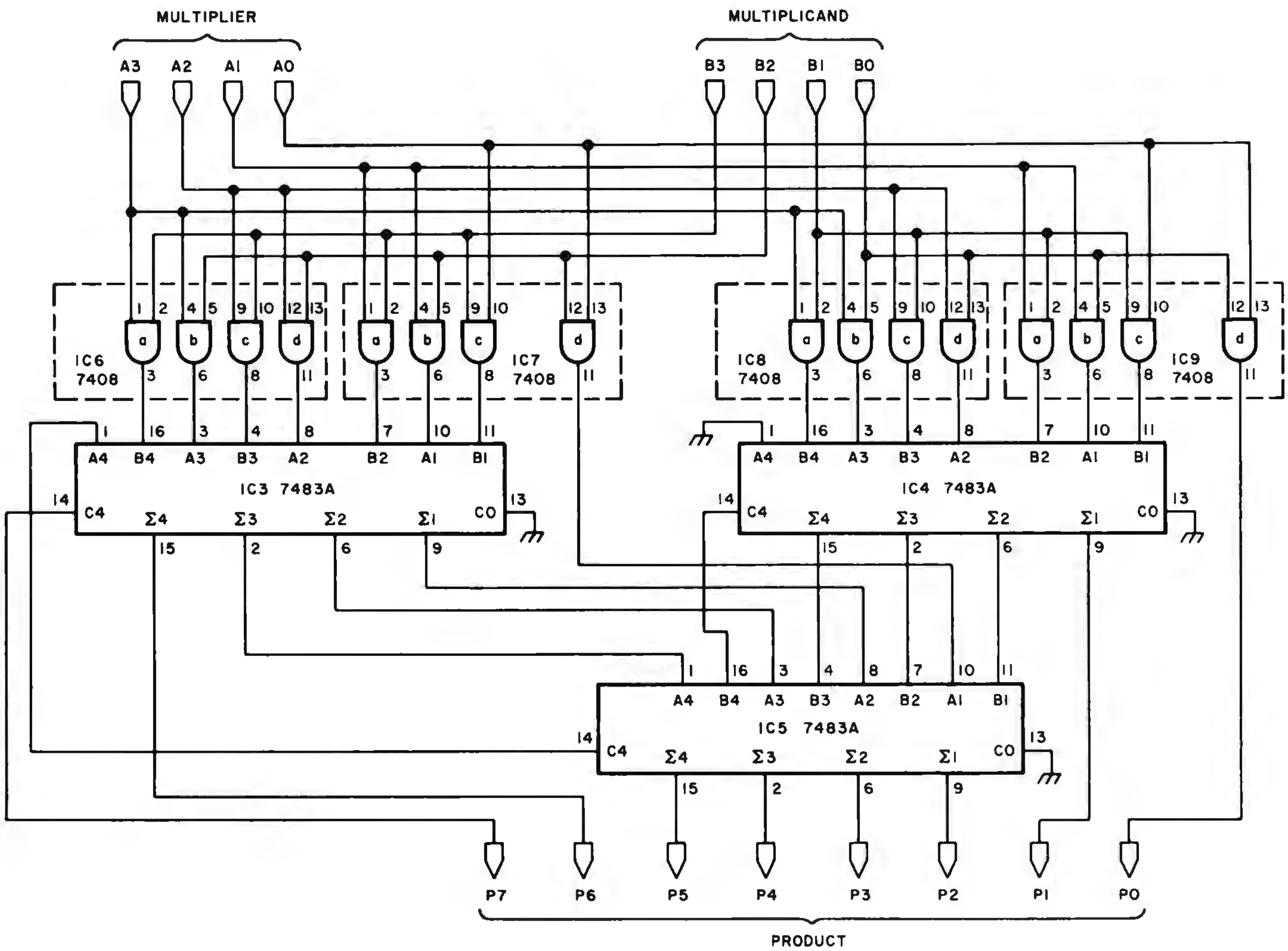


Figure 2: Inexpensive 4 by 4 multiplier that has a higher number of integrated circuits and uses more power than the circuit in figure 1.

The circuit in figure 3 shows one of many possible variations along these lines. This 4 by 5 multiplier uses inverters and NOR gates on the inputs instead of the AND gates of figure 2. Desirable input buffering is thus provided at about the same dollar cost, but requires more packages and power.

Clockless division circuits are more complicated than multiplication circuits. One way to obtain a 4 bit quotient from a 7 bit dividend and a left-adjusted 4 bit divisor is shown in figure 4. A subtraction is performed at each stage by complementing the divisor and adding, while forcing a carry input to the adder. If the difference is negative, as indicated by a 0 carry output, a 4 bit 1 out of 2 multiplexer is set to transmit the minuend to the next stage. If the difference

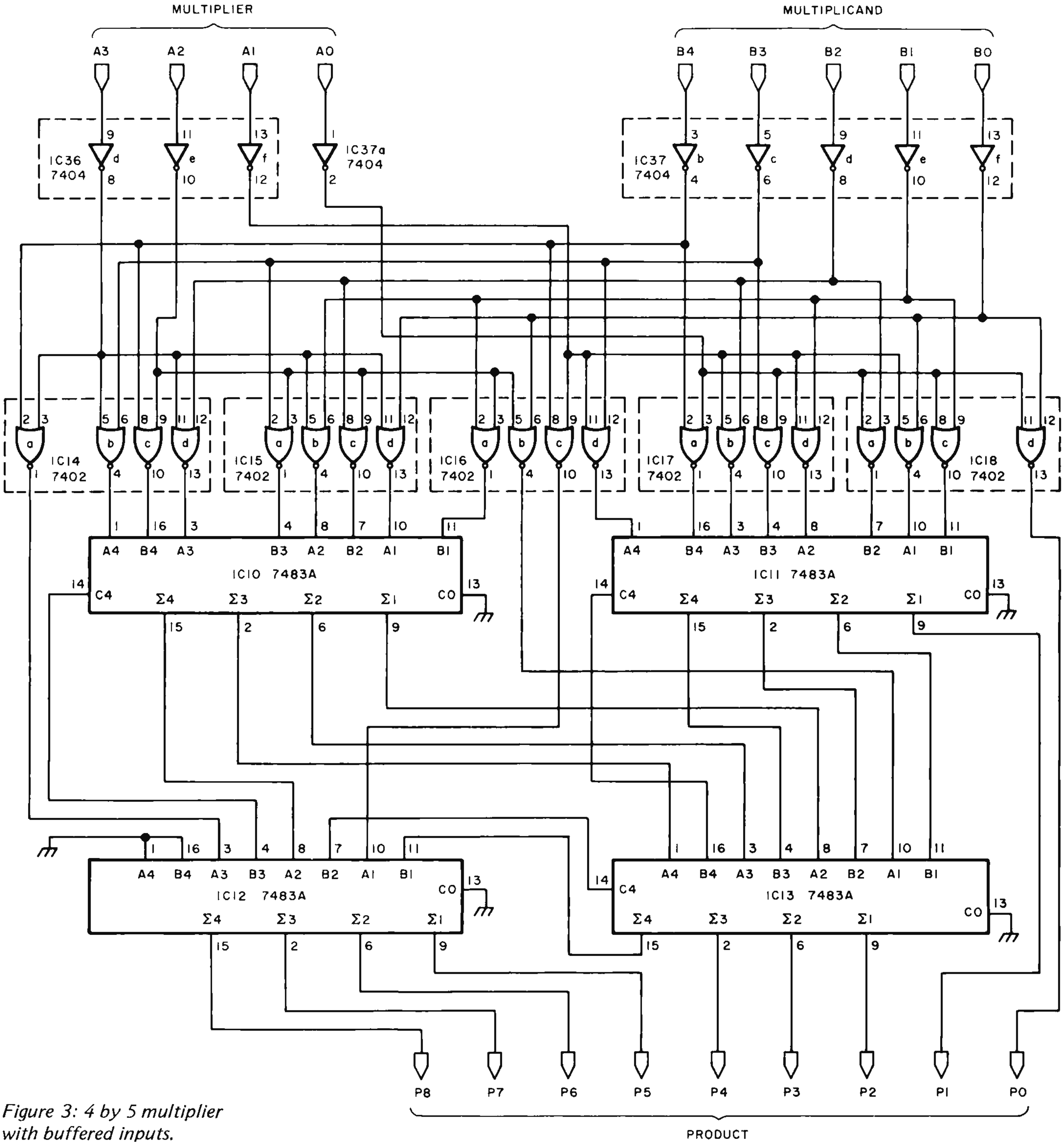


Figure 3: 4 by 5 multiplier with buffered inputs.

Osborne & Associates

BUSINESS SOFTWARE FOR MICROCOMPUTERS

Osborne & Associates is publishing its business systems in book form. These systems represent five years of development and testing by O&A programmers, and the books include more than a year's worth of extensive and detailed documentation.

What systems are we selling?

1) **PAYROLL WITH COST ACCOUNTING** — available now
2) **ACCOUNTS PAYABLE AND ACCOUNTS RECEIVABLE** — available now
3) **GENERAL LEDGER** — orders now being accepted

Each book sells for $15.00, and includes source listings in Wang BASIC, program and system documentation, and user's manual. Each is a complete package by itself, or all three may be implemented together to form a complete system with interdependent files.

And if Wang BASIC won't work, or you don't know programming, or you'd rather not key in thousands of words of source code, take a look at the list of consultants who have adopted O&A programs, converted them to run on many popular systems, and are waiting to hear from you.

CP/M CBASIC: GOOD NEWS for CONSULTANTS, COMPUTER STORES AND SOFTWARE HOUSES

Osborne & Associates is converting its business software from Wang BASIC - as it was originally published - to CP/M CBASIC, which runs on many floppy disk-based microcomputer systems. We will only sell the CP/M magnetic surface tc consultants, computer stores and software houses. Osborne & Associates prefer to write and sell books, not customize the programs or answer the end user's questions. The disk for each book sells for $250.00. Once you buy the floppy disk you can copy it, resell it, change it or use it. We place no restriction on the magnetic surface; we copyright only the printed word in our books. CBASIC Payroll is available now. All three systems are scheduled to be available by the end of 1978; call or write for the exact availability of each system.

If you are an end user interested in the CBASIC programs, write or call us. We will put you in touch with your closest dealer.

DIGITAL GROUP:
John Musgrove
MUSGROVE ENGINEERING
9547 Kindletree Drive
Houston, TX 77040
(713) 466-3486

DIGITAL GROUP Z80, Oasis BASIC:
Charles Finn
THE SYSTEMS GROUP
2993 Boring Ridge Drive
Decatur, GA 30034
(404) 289-8969

IBM 5110:
Mark Sherman
DATA WORKS
35 E. Wacker Drive, Suite 1843
Chicago, IL 60601
(312) 726-2572

MITS 4.1 BASIC:
William K. Haines
ANACOM GENERAL CORPORATION
1160 E. Ash Avenue
Fullerton, CA 92631
(714) 992-0223

NORTH STAR:
Alan P. Hald
BYTE SHOP OF ARIZONA
813 N. Scottsdale Road
Tempe, AZ 85281
(602) 967-1428

TRS-80:
Don French
TANDY COMPUTERS
700 One Tandy Center
Fort Worth, TX 76102
(817) 390-3011

MICROSOFT disk BASIC:
Dan Kindred
GNAT COMPUTERS, INC
7895 Convoy Court
San Diego, CA 92111
(714) 560-0433

CP/M CBASIC:
OSBORNE & ASSOCIATES, INC.
630 Bancroft Way
Berkeley, CA 94702
(415) 548-2805

MICROPOLIS, PROCESSOR TECH, NORTHSTAR:
David Price
3901 Victoria Lane
Midlothian VA 23113

Wang BASIC on floppy disk:
Richard Armour
ATLANTIC COMPUTER & CONSULTING
1104 Sparrow Road
Chesapeake, VA 23325
(804) 764-3412

TRS-80:
Jim Smith
READY CORPORATION
2502 Calle Madras
Pleasanton, CA 94566
(415) 462-4381

Osborne & Associates, Inc.
P.O. Box 2036, dept. 115
Berkeley, CA 94702 USA
(415) 548-2805
TWX 910-366-7277

NAME

ADDRESS

CITY STATE

ZIP PHONE

☐ I am a dealer interested in purchasing CBASIC programs for resale. Please send description and order information for:

☐ I am an end user interested in running CBASIC programs. Please notify me of the closest dealer for:

☐ CBASIC Payroll with Cost Accounting
☐ CBASIC Accounts Payable/Receivable
☐ CBASIC General Ledger

TITLE	PRICE	QUANTITY	AMOUNT
09-8 Payroll with Cost Accounting	$15.00		
13-8 Accounts Payable/Receivable	$15.00		
20-9 General Ledger	$15.00		

SHIPPING (Shipping for large orders to be arranged)
☐ All foreign orders $3.00 per book for airmail
☐ $0.35 per book 4th class (allow 2-3 weeks in the U.S.)
☐ $0.75 per book UPS in the U.S. (allow 10 days)
☐ $1.50 per book special rush shipment by air in the U.S

California residents tax

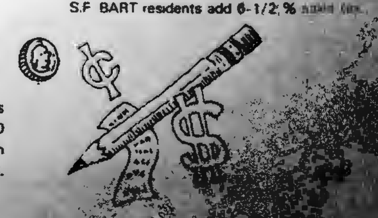
shipping

TOTAL AMOUNT ENCLOSED

Please send the following information
☐ Becoming an O&A dealer
☐ School discounts
☐ List of foreign distributors
☐ Other O&A publications

Calif. residents add 6% sales tax.
S.F. BART residents add 6-1/2% sales tax.

Payment in advance must be enclosed for purchases of up to $70.00. Invoicing U.S. purchases over $70.00 available upon approval of your account. All foreign orders must be prepaid in U.S. dollars drawn on a U.S. bank.



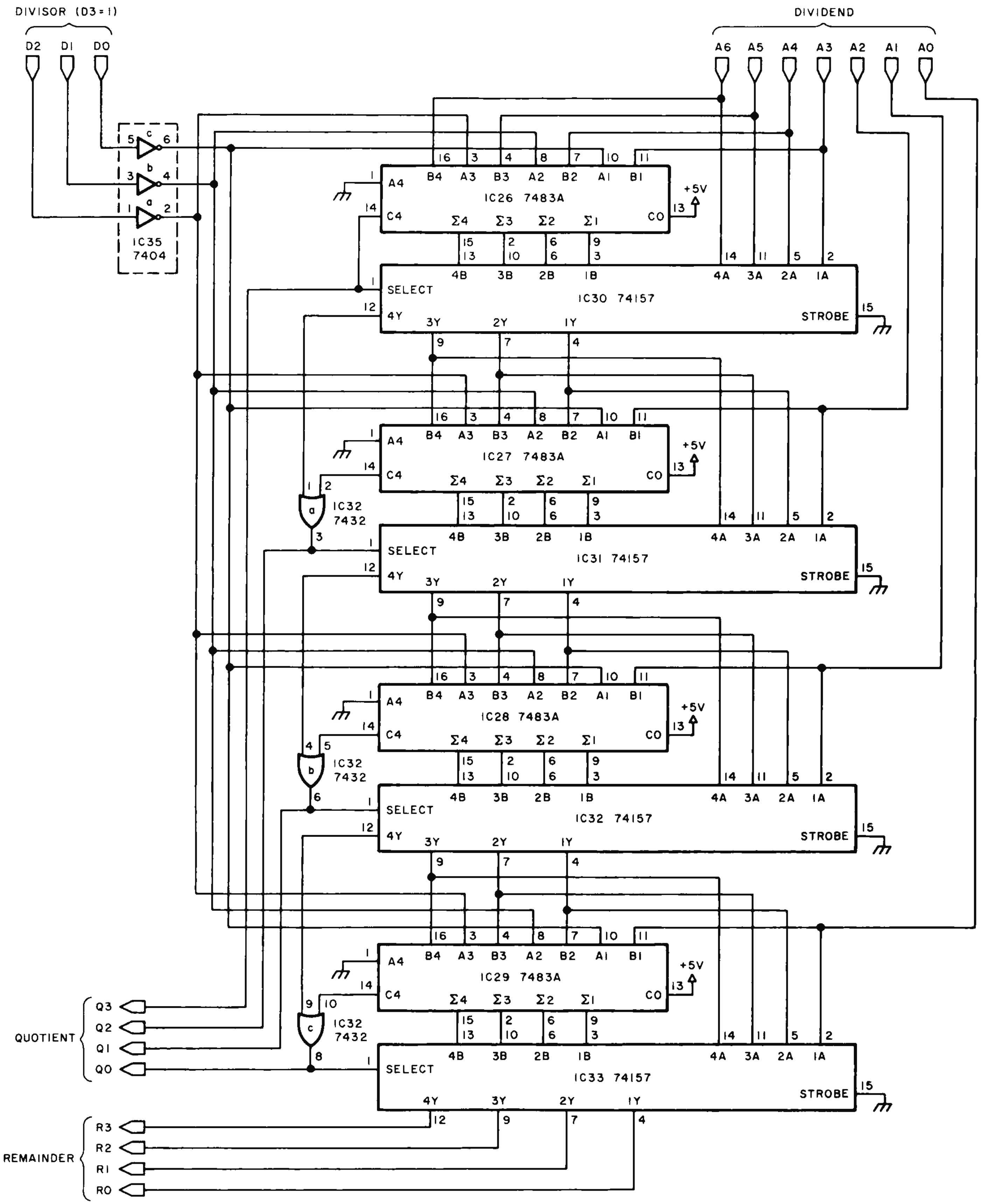
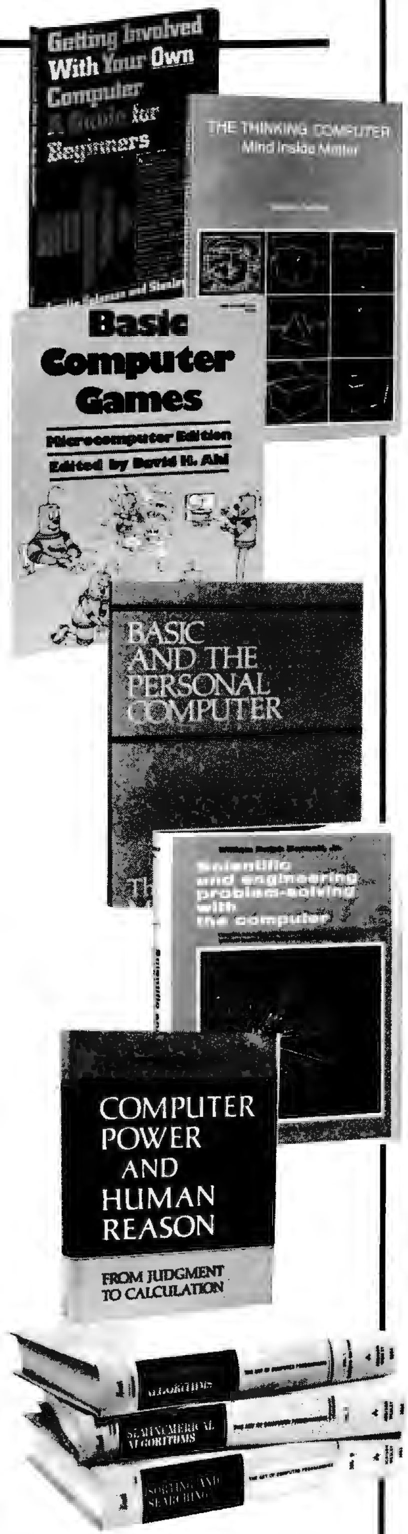


Figure 4: Simple but slow clockless division circuit.

The Microcomputer Bookstore BITS T.M.

25 Route 101 West, Peterborough NH 03458

Best Sellers



* __**Getting Involved With Your Own Computer** by Leslie Solomon and Stanley Veit answers the questions: "What can small computers do? Which is best for my purposes?" Whether your interest is business applications, word processing, education, security, etc., this lucid text will bring you in touch with an exciting new world destined to affect us all. $5.95.

* __**The Thinking Computer: Mind Inside Matter** by Bertram Raphael. Artificial intelligence, or AI, is the branch of computer science concerned with making computers "smarter." It is a growing, vital field that is, unfortunately, the subject of much popular misunderstanding. This book is a lucid introduction to AI that does much to overcome this misunderstanding. With a minimum of technical jargon, it discusses the capabilities of modern digital computers and how they are being used in contemporary AI research. It discusses the progress of AI, the goals, and the variety of current approaches to making the computer more intelligent. $6.95.

* __**BASIC Computer Games: Microcomputer Edition** edited by David H Ahl. Here are 102 classic computer games, every one in standard microcomputer BASIC; every one complete with large, legible listing, sample run and descriptive notes. All the classics are here: Super Star Trek (one of the most challenging versions anywhere), Football (two versions), Blackjack, Lunar Lander (three versions), Tic Tac Toe, Nim, Life and Horserace. This revision of **101 BASIC Computer Games** is a real must, even if you own the original. $7.50.

* __**BASIC and the Personal Computer** by Thomas A Dwyer and Margot Critchfield. A fascinating book covering many areas of interest to the personal computer user. After giving an in-depth course in BASIC, which can be covered in 8 hours, the book discusses microcomputer hardware, graphics, word processing, sorting, simulation and data structures. This is an easy to read text that is useful for the beginner and informative for the advanced user. 438 pp. $12.95.

* __**Scientific and Engineering Problem Solving With The Computer** by William Ralph Bennett Jr is one of the most exciting books we've seen in years. Besides teaching BASIC, this lively, lucid book presents a wealth of imaginative and unusual applications programs taken from many disciplines (A sample exercise: "Using the algorithm in the text with the pair-correlation matrix from Hamlet, compute the most probable diagram path which starts with the letter T"). The exercises run the gamut from random processes to the dynamics of motion, from entropy in language to the Watergate problem. You'll discover BASIC applications in lasers and in the Fourier series and the law (!). In its diversity and elegant style, it ranks with Donald Knuth's works as a milestone in the art of computing. Hardcover $19.95.

* __**Computer Power and Human Reason** by Joseph Weizenbaum. This book is one which should be purchased or read for several reasons. If you're presently a programmer by trade or skill, you'll see a philosophy of computer use and abuse propounded. It's genuinely interesting, and definitely provocative if you reference the storm of letters, counter letters and counter counter letters which this book produced in the Association for Computing Machinery's *SIGART* newsletters during 1976. If you're a novice to the field, the tutorial and explanatory chapters of this book, which are aimed at the layperson, will serve as an excellent background source which is also eminently readable. This includes an excellent and low level explanation of what an algorithm is, and how computers go about executing effective algorithms. $5.95.

Praised by many critics as the best books in their field, **The Art of Computer Programming, Volumes I, II and III,** are part of a projected seven volume omnibus survey of computer science now being completed by Donald E Knuth.

* __Volume I, **Fundamental Algorithms,** begins with a thorough discussion of the mathematics used in computer programming, followed by a treatment of information structures, stacks, arrays, linked lists, dynamic storage allocation, and trees. 634 pp. $21.95.

* __Volume II, **Seminumerical Algorithms,** is concerned with random numbers, statistical tests, random sequences, as well as arithmetic (floating point and multiple precision), polynomials, and rational arithmetic. 624 pp. $21.95.

* __Volume III deals with **Searching and Sorting,** and as the name implies, the emphasis is on algorithms for sorting, including combinatorial properties of permutations, internal sorting, optimum sorting, and external sorting. Also included is a section on sequential searching, hashing, digital searching, and more. 722 pp. $21.95.

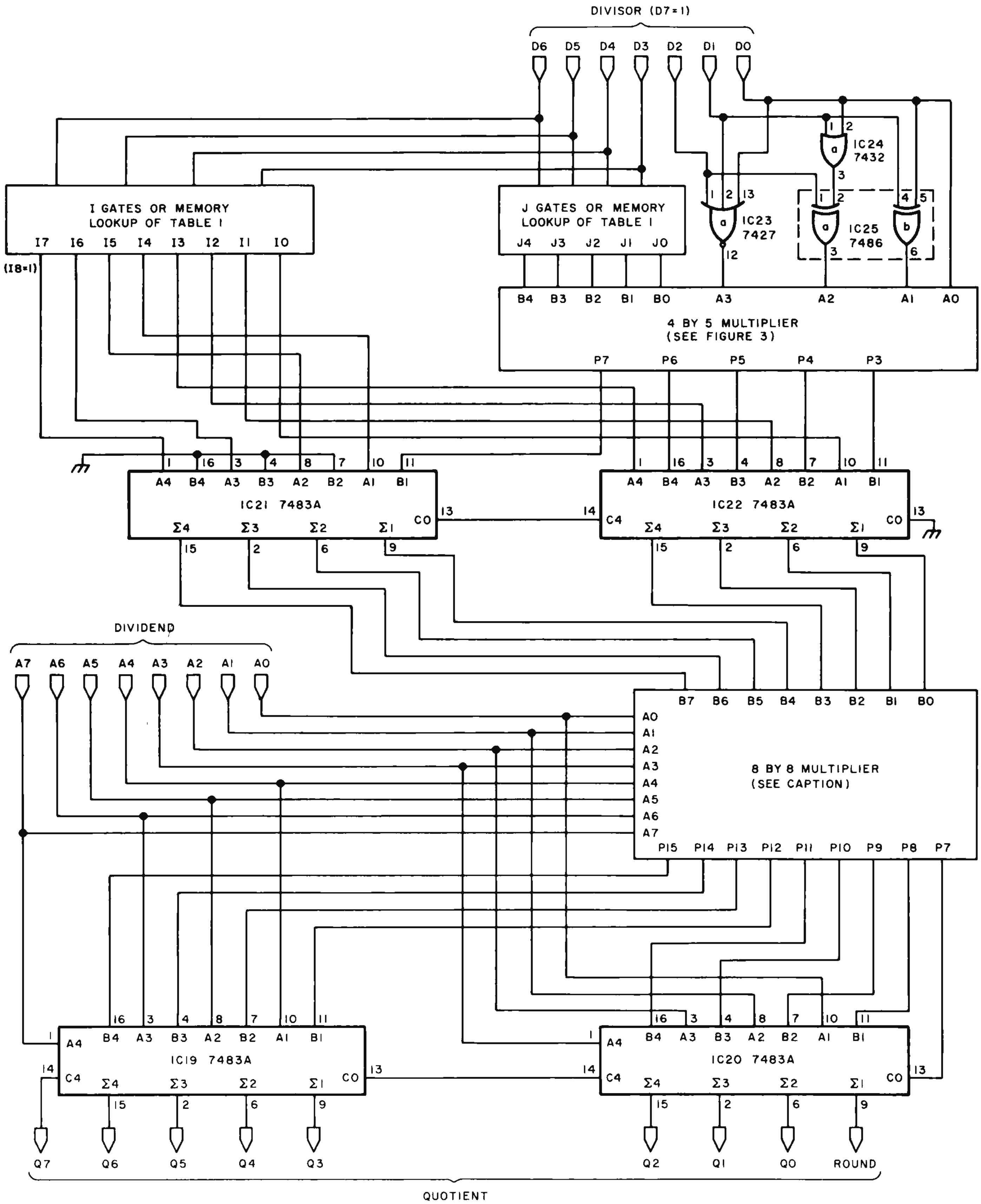
A hypothetical assembly language called MIX was developed by the author to illustrate programming examples throughout the series. MIX is easily convertible to other assembly languages.

Professor Knuth writes with style and wit. This classic work belongs on the reference shelf of everyone seriously interested in computer science.

For your convenience in ordering, please use this page plus the order form on page 138. You may photocopy this page.

DIAL YOUR CHARGE CARD ORDERS TOLL-FREE 1-800-258-5477.

Circle 35 on inquiry card.


DIVISOR (D7=1)
D6 D5 D4 D3 D2 D1 D0
IC24 7432
IC23 7427
IC25 7486
I GATES OR MEMORY LOOKUP OF TABLE 1
I7 I6 I5 I4 I3 I2 I1 I0
(I8=1)
J GATES OR MEMORY LOOKUP OF TABLE 1
J4 J3 J2 J1 J0
4 BY 5 MULTIPLIER (SEE FIGURE 3)
B4 B3 B2 B1 B0 A3 A2 A1 A0
P7 P6 P5 P4 P3
IC21 7483A
IC22 7483A
DIVIDEND
A7 A6 A5 A4 A3 A2 A1 A0
8 BY 8 MULTIPLIER (SEE CAPTION)
B7 B6 B5 B4 B3 B2 B1 B0
P15 P14 P13 P12 P11 P10 P9 P8 P7
IC19 7483A
IC20 7483A
Q7 Q6 Q5 Q4 Q3 Q2 Q1 Q0 ROUND
QUOTIENT

is positive, the multiplexer transmits this difference. The quotient bits are the same bits that set the multiplexers. This method is straightforward and provides a remainder, but has the disadvantage of being relatively slow.

A more elegant clockless divider makes use of the relation $1/(1-r) \approx 1+r$, provided the absolute value of r is much less than 1. The error in this estimate is r^2, as can be seen by multiplying each side of the approximate equality by $(1-r)$. Suppose an 8 bit dividend $(A = a_7a_6a_5a_4a_3a_2a_1a_0)$ is to be divided by an 8 bit left-adjusted divisor $(D = d_7d_6d_5d_4d_3d_2d_1d_0)$ to yield an 8 bit integer quotient Q. Let:

$$r = (2^3-d_2d_1d_0)/[2^3(d_7d_6d_5d_4d_3+1)].$$

This means that:

$$D = (d_7d_6d_5d_4d_3+1)(2^3)(1-r)$$
$$Q = 2^7 (A)/[(d_7d_6d_5d_4d_3+1)(2^3)(1-r)],$$
$$Q \approx 2^4(A)(1+r)/(d_7d_6d_5d_4d_3+1).$$

Since d_7 equals 1, r is less than $2^3/(2^3 \times 2^4)$ or 2^{-4}, and r^2 is less than 2^{-8}. Q in this approximation is accurate to at least eight bits. Let:

$$I = 2^{12}/(d_7d_6d_5d_4d_3+1)$$

and

$$J = 2^{12}/(d_7d_6d_5d_4d_3+1)^2$$

each rounded to the nearest integer. Then Q is approximately equal to $2^{-8}(A)[I+(2^3-d_2d_1d_0)(2^{-3})(J)]$ and the division problem has been reduced to addition and multiplication once I, J and $(2^3-d_2d_1d_0)$ have been determined.

This last quantity is easily derived from four simple gates, as illustrated in the complete divider of figure 5. The quantities I and J are listed in table 1 for all possible values of $d_7d_6d_5d_4d_3$. These are found to 9 bit and 5 bit accuracy, respectively, to insure 8 bit accuracy in Q after the intermediate

Figure 5: Faster 8 bit clockless divider. The 4 by 5 multiplier is the circuit of figure 3. The I and J values are obtained from a set of gates or a lookup table. The results being looked for are the values given in table 1. The 8 by 8 multiplier is the only device not previously discussed. The multiplier is composed of bit slices; the theory behind the multiplication circuits can be found in the TTL Data Book for Design Engineers, *published by Texas Instruments Inc. The device number is SN54LS275, found on page 7-391 of the 1976 edition.*

Table 1: List of I and J values for 8 bit divider circuit of figure 5.

UPPER D	I	J
$d_7d_6d_5d_4d_3$	$i_8i_7i_6i_5i_4i_3i_2i_1i_0$	$j_4j_3j_2j_1j_0$
1 0 0 0 0	1 1 1 1 0 0 0 1 0	1 1 1 0 0
1 0 0 0 1	1 1 1 0 0 0 1 1 1	1 1 0 0 1
1 0 0 1 0	1 1 0 1 0 1 1 1 1	1 0 1 1 1
1 0 0 1 1	1 1 0 0 1 1 0 1 0	1 0 1 0 0
1 0 1 0 0	1 1 0 0 0 0 1 1 0	1 0 0 1 1
1 0 1 0 1	1 0 1 1 1 0 1 0 0	1 0 0 0 1
1 0 1 1 0	1 0 1 1 0 0 1 0 0	0 1 1 1 1
1 0 1 1 1	1 0 1 0 1 0 1 0 1	0 1 1 1 0
1 1 0 0 0	1 0 1 0 0 1 0 0 0	0 1 1 0 1
1 1 0 0 1	1 0 0 1 1 1 0 1 1	0 1 1 0 0
1 1 0 1 0	1 0 0 1 0 1 1 1 1	0 1 0 1 1
1 1 0 1 1	1 0 0 1 0 0 1 0 1	0 1 0 1 0
1 1 1 0 0	1 0 0 0 1 1 0 1 0	0 1 0 1 0
1 1 1 0 1	1 0 0 0 1 0 0 0 1	0 1 0 0 1
1 1 1 1 0	1 0 0 0 0 1 0 0 0	0 1 0 0 1
1 1 1 1 1	1 0 0 0 0 0 0 0 0	0 1 0 0 0

Table 2: Power connections for integrated circuits used in figures 1 thru 5.

Power Wiring Table

Number	Type	+5 V	Gnd
IC1	74284	16	8
IC2	74285	16	8
IC3	7483A	5	12
IC4	7483A	5	12
IC5	7483A	5	12
IC6	7408	14	7
IC7	7408	14	7
IC8	7408	14	7
IC9	7408	14	7
IC10	7483A	5	12
IC11	7483A	5	12
IC12	7483A	5	12
IC13	7483A	5	12
IC14	7402	14	7
IC15	7402	14	7
IC16	7402	14	7
IC17	7402	14	7
IC18	7402	14	7
IC19	7483A	5	12
IC20	7483A	5	12
IC21	7483A	5	12
IC22	7483A	5	12
IC23	7427	14	7
IC24	7432	14	7
IC25	7486	14	7
IC26	7483A	5	12
IC27	7483A	5	12
IC28	7483A	5	12
IC29	7483A	5	12
IC30	74157	16	8
IC31	74157	16	8
IC32	74157	16	8
IC33	74157	16	8
IC34	7432	14	7
IC35	7404	14	7
IC36	7404	14	7
IC37	7404	14	7

steps. I and J may be determined for a given divisor by a lookup process or by a suitable arrangement of gates. Of course, the entire division may be performed by looking up the inverse of the 8 bit divisor and then multiplying, but the method described here uses one eighth the memory space and only slightly more circuitry.

This process uses the inverse of a small number to find the inverse of a larger number, and so suggests a procedure for handling the division of numbers of arbitrary size. Such multiple-precision calculations could be performed by expanding the kind of hardware described here, or by an iterative software routine. Remainders are not directly available from this circuit, and must be obtained by subtracting the product of the quotient and divisor from the dividend.

The requirement that the divisor be left-adjusted is something of a nuisance; dividers generally need a restriction of this sort to keep the calculation in range of the hardware capability. Methods exist to make this adjustment and the subsequent adjustment required in the quotient without the use of clock pulses. These cumbersome circuits will not be described here.

Is any of this useful to the small systems owner? Most microprocessors do not have multiplication or division instructions. Products and quotients are obtained through time-consuming subroutines. Computer generated music or animated video displays may not permit sufficient computation time. Such real time outputs would be feasible if the fast circuits described here were incorporated into an external arithmetic unit and accessed through the input/output (IO) ports of the microcomputer. Then a division would be performed by the following:

- Output the divisor to the external arithmetic unit divider.
- Output the dividend to the external arithmetic unit divider.
- Input the quotient from the external arithmetic unit divider.
- Input the remainder (if desired) from the arithmetic unit divider.

Holding registers are required for the divisor and dividend. If 8 bit arithmetic is used, the entire calculation can be performed easily in the time taken by the input and output instructions.■

Editor's Note:

*These circuits are theoretical. They have been designed but not implemented by the author. . . .*RGAC.

Creating a Chess Player Part 3:

Chess 0.5 (continued)

Larry R Atkin
Health Information Services
542 Michigan Av
Evanston IL 60202

Peter W Frey
Dept of Psychology
Northwestern University
Evanston IL 60201

This month we conclude the listing and commentary of Chess 0.5 begun last issue. The program was written by Larry Atkin, who is coauthor with David Slate of the world championship chess program, Chess 4.6. The program is readily adaptable to personal computers having Pascal systems such as the UCSD Pascal project software. Part 4 concludes the series with a discussion of chess strategy and tactics.

Listing 1: The second half of Chess 0.5, written in Pascal. This portion of the program covers evaluation of terminal nodes, the look-ahead procedure and user commands *(listing 1 continued on page 146).*

```
PROCEDURE EVALU8;                          (* EVALUATE CURRENT POSITION *)

VAR
  INTV : TV;                               (* SCORE *)


  FUNCTION EVKING                          (* EVALUATE KING *)
    (A:RS;                                 (* KING BIT BOARD *)
     B:RS):TV;                             (* FRIENDLY PAWN BIT BOARD *)

  VAR
    INTS : TS;                             (* SCRATCH *)
    INRS : RS;                             (* SCRATCH *)
    INTV : TV;                             (* SCRATCH *)

  BEGIN
    ANDRS(INRS,A,CORNR);
    IF NULRS(INRS) THEN                    (* KING NOT IN CORNER *)
      INTV := 0
    ELSE
      INTV := FKSANQ;                      (* KING SAFELY IN CORNER *)

    INRS := A;
    IF NXTTS(INRS,INTS) THEN
    BEGIN
      ANDRS(INRS,ATKFR[INTS],B);           (* FIND PAWNS NEXT TO KING *)
      INTV := INTV + CNTRS(INRS)*FKPSHD;
                                           (* CREDIT EACH CLOSE PAWN *)
    END;

    EVKING := INTV;                        (* RETURN KING SCORE *)
  END;  (* EVKING *)


  FUNCTION EVMOBL                          (* EVALUATE MOBILITY *)
    (A,B:TP):TV;                           (* PIECE TYPES TO EVALUATE *)

  VAR
    INRS : RS;                             (* SCRATCH *)
    INTS : TS;                             (* SCRATCH *)
    INTV : TV;                             (* SCRATCH *)

  BEGIN
    IORRS(INRS,TPLOC[A],TPLOC[B]);         (* MERGE PIECE TYPES *)
    INTV := 0;                             (* INITIALIZE COUNT *)
    WHILE NXTTS(INRS,INTS) DO              (* COUNT ATTACKS *)
      INTV := INTV + CNTRS(ATKFR[INTS]);
    EVMOBL := INTV;                        (* RETURN TOTAL ATTACKS *)
  END;  (* EVMOBL *)


  FUNCTION EVPAWN                          (* EVALUATE PAWNS *)
    (A:RS;                                 (* LOCATION OF PAWNS *)
     B:TE;                                 (* PAWN FORWARD DIRECTION *)
     C:TR):TV;                             (* PAWN HOME RANK *)

  VAR
    INRS : RS;                             (* SCRATCH *)
    IMRS : RS;                             (* SCRATCH *)
    INTS : TS;                             (* SCRATCH *)
    INTV : TV;                             (* SCRATCH *)

  BEGIN
    SFTRS(INRS,A,S1);
    ANDRS(INRS,INRS,A);                    (* BIT SET FOR SIDE BY SIDE *)
    INTV := CNTRS(INRS)*FPFLNX;            (* SCORE PHALANX *)

    SFTRS(INRS,A,B1);
    ANDRS(INRS,INRS,A);                    (* BIT SET FOR PAWN DEFENSE *)
    INTV := INTV + CNTRS(INRS)*FPCONN;     (* CREDIT CONNECTED PAWNS *)

    SFTRS(INRS,A,B2);
    ANDRS(INRS,INRS,A);
    INTV := INTV + CNTRS(INRS)*FPCONN;     (* AND OTHER CONNECTED PAWNS *)

    SFTRS(INRS,A,B);                       (* MOVE FORWARD *)
    NOTRS(IMRS,TPLOC[MT]);                 (* OCCUPIED SQUARES *)
    ANDRS(INRS,INRS,IMRS);                 (* BLOCKED PAWNS *)
    INTV := INTV - CNTRS(INRS)*FPBLOK;     (* PENALIZE BLOCKED PAWNS *)

    CPYRS(INRS,A);
    WHILE NXTTS(INRS,INTS) DO              (* FOR EACH PAWN *)
      INTV := INTV +(ABS(ORD(C)-ORD(XTSR[INTS])))*FPADCR[XTSF[INTS]];
                                           (* CREDIT PAWN ADVANCEMENT *)

    EVPAWN := INTV;                        (* RETURN PAWN SCORE *)
  END;  (* EVPAWN *)
```

Evaluating Terminal Positions

Another important aspect of any chess program is the function which provides a static evaluation of terminal positions in the look-ahead tree. In the present program, this routine also doubles as a preliminary scoring function for sorting moves at the first ply, at the beginning of the look-ahead search. Since the evaluation function is used repetitively in the search, efficiency demands that it be carefully engineered. We have left this task as an exercise for the reader. Our function presently includes only a few basic essentials.

The most important feature is material. We employ essentially the same function for this that is used by Chess 4.5. A trade-down bonus is also incorporated, ie: trade pieces but not pawns when ahead in material. A second feature which is considered is piece mobility. The mobility of Knights and Bishops is weighted more heavily than that for Rooks and Queens. Special credit is given to a King which is located in one of the four corner squares in each corner of the board, ie: 16 squares total. This encourages early castling. Pawn structure is considered by providing a bonus for advancing the pawns in the four center files, for having a pawn near the King, and for having a pawn adjacent to or defended by another pawn. This indirectly penalizes isolated or backward pawns. There is a direct penalty if the square in front of a pawn is occupied. The position of the Rooks is considered by providing a bonus for placing a Rook on the seventh rank and for attacking another Rook of the same color (ie: doubled Rooks). The executive routine for these assessments is EVALU8.

The Look-Ahead Procedure

The look-ahead procedure is controlled by an executive routine called SEARCH. Several subprocedures are also defined which handle specific tasks. NEWBST keeps track of the move which is currently thought to be best, and dynamically reorders the moves at the first ply level each time a new best-move is selected. MINMAX determines whether the move under consideration will produce an α-β cutoff. SCOREM is called into action when the program can find no legal moves at a node. It determines whether the position should be scored as a checkmate or as a stalemate. SELECT is responsible for move ordering at each node. It determines whether there are any more moves to be searched and if so, makes sure that they are generated in the correct order (ie: captures, killers, castling moves, and then the remaining moves).

SEARCH incorporates a number of important features which make the look-ahead search more efficient. These include staged move generation, preliminary ordering scores, setting a narrow α-β window at the beginning of the search, conducting the search in an iterative fashion, and dynamically recording moves at the first ply as the search proceeds. Because of these features, the full-width search takes a long time instead of taking forever.

User Commands

For the user's convenience, the program should be able to respond to a few simple commands. Inputs to the program are processed by a lengthy routine, READER, which has many component subprocedures. The translation of the input string is handled by a group of routines: RDRERR, RDRGNT, RDRSFT, RDRCMP, RDLINE, RDRMOV and RDRNUM. Each of the commands is executed by a separate routine.

When the human player wishes to terminate the game before it has reached its conclusion (eg: when he is hopelessly lost and does not want to stay around to be crushed), he can simply type an END command and the ENDCMD routine will terminate the program. If the user simply wishes to start a new game, he can type INIT and the INICMD routine will set up for a new game.

If the user would like to set up a specific position from the previous game or some other game, he can call the BOACMD routine, which will set up any position he desires. To use this instruction, the pieces are designated in the standard way (eg: K, Q, R, B, N and P) and the colors are designated by L for light and D for dark. The board is described by starting at the lower lefthand corner and listing, row by row, the 64 squares. Numbers are used to represent consecutive empty squares. The command to set up the position after 1. P-K4, P-K4, 2. N-KB3, N-QB3 is: BOARD, LRNBQKB1 RPPPP1PPP5N24P34DP33N4PPPP1PPPR1B QKBNR.

If the human player is lazy or simply wishes to test the program, he or she can type GO and the machine will select a move. By repeatedly typing GO the user can sit back and watch the machine play against itself. The routine that handles this is GONCMD. To specify a value for selected program parameter variables, the player can use LETCMD. For example, the amount of time the machine spends calculating a move can be controlled by specifying a limit for the number of nodes to be searched. The command LET FNODEL = 1000 will cause the machine to set a target value of 1000 for the number of nodes to be searched. In this case it will not start another iteration if it has already searched 1000 nodes. If the user is confused about the current board configuration, the command PRINT will activate PRICMD which calls PRINTB for a representation (8 by 8 array) of the board. For diagnostic purposes the user can also ask for other information. The routine PAMCMD is activated by PB and provides an 8 by 8 attack map for each of the 64 squares. The routine POPCMD is activated by PO and gives information concerning the side to move (White or Black), the en passant status after the last move, the present castle status and the move number. If the user types PM, the routine PMVCMD will provide a list of all moves which are legal for the side to move in the current position. The command PL activates PLECMD which prints the value of a designated variable; for example, the user can determine the present limit for the number of nodes to be searched by typing PL FNODEL.

The user also has control over several switches. He can ask the machine to repeat (echo) each entry, to pause after 20 lines of output, and to reply automatically each time the opponent enters a move. These switches are set by the switch commands (eg: SW EC OFF), and are processed by SWICMD. If the user wishes to manually alter one or more of the status conditions

(eg: side to move, move number, en passant, castling), this can be done by activating STACMD.

Notes on Notation

The program also processes standard chess notation. This is not strictly necessary. Many programs use their own convention for entering and reporting moves. A common procedure is to denote the squares using a number (1 through 8) for each row and a letter (A through H) for each column. A move is defined by listing the present square of the piece and then the destination square. For example, the common opening move, P-K4, would be E2E4. Moving the White Knight on the kingside from its original square to KB3 would be G1F3. This convention works nicely but it forces an experienced chess player to learn a new system. Most would prefer standard chess notation.

Because there are multiple ways to express the same move in standard notation, the translation routine needs to be fairly sophisticated. Consider a position in which the White Queen's Rook is on its original square and the neighboring Knight and Bishop have been moved. A move which places the Rook on the Queen Bishop file can be designated as R-B1, R-QB1, R/1-B1, R/1-QB1, R/R1-B1, or R/R1-QB1. It is important that the program recognize that each of these character strings represents the same move. How is this done?

One way is to have the machine generate a list of all legal moves and then compare each of these with the move entered by the player. If his move matches one on the list, that move is noted. The rest of the list is then checked and if no more matches are found, the noted move is assumed to be the correct one. If no match is found, the machine prints "illegal move." If a second match is found (eg: P-B3 matches both P-KB3 and P-QB3), the machine prints "ambiguous move." The process of translating the opponent's move into machine compatible form and checking its legality or ambiguity is done by YRMOVE. The process of translating the machine's move into standard notation is handled by MYMOVE. Both of these procedures call MINENG, which is responsible for constructing the appropriating character strings.

Final Thoughts

This completes our listing of our demonstration chess program. Despite the program's length, there are many desirable features which have been omitted. The reader with an interest in chess and programming should use this listing as a starting point for developing a program. The time required for move calculation can be reduced by writing machine dependent code for some of the frequently used routines. There are also features which can be added to improve the level of play.

One useful addition would be an opening library. An effective technique for this is described by Slate and Atkin in their chapter in *Chess Skill in Man and Machine* (P W Frey, editor, Springer-Verlag, New York, 1977). An opening library provides the user with a challenging set of opening moves and directs the game into situations which are familiar to the experienced chess player. By including various options at the early choice points and using a random selection procedure, the programmer can insure that the machine will not always select the same move sequence. The programmer can also give the user the option of specifying a particular opening against which he would like to practice. For important matches, the programmer can prepare surprise openings for the machine in order to gain a psychological edge on the opponent.

Text continued on page 157

Listing 1, continued from page 140:

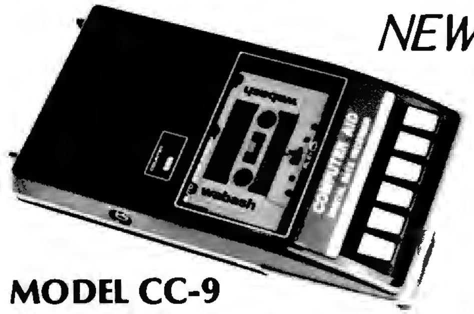
```
  FUNCTION EVROOK                              (* EVALUATE ROOKS *)
    (A:RS;                                     (* ROOK LOCATIONS *)
     B:RS):TV;                                 (* SEVENTH RANK *)

  VAR
    INTV : TV;                                 (* SCRATCH *)
    INTI : TI;                                 (* SCRATCH *)
    INTS : TS;                                 (* SCRATCH *)
    INRS : RS;                                 (* SCRATCH *)

  BEGIN
    INTV := 0;                                 (* INITIALIZE *)
    INRS := A;
    IF NXTTS(INRS,INTS) THEN                   (* LOCATE FIRST ROOK *)
    BEGIN
      ANDRS(INRS,A,ATKFR[INTS]);
      IF NOT NULRS(INRS) THEN                  (* ROOK ATTACKS FRIENDLY ROOK *)
        INTV := INTV + FRDUBL;                 (* GIVE DOUBLED ROOK CREDIT *)
    END;

    ANDRS(INRS,A,B);                           (* ROOKS ON SEVENTH *)
    INTI := CNTRS(INRS);
    EVROOK := INTV + INTI*INTI*FRK7TH;  (* CREDIT ROOKS ON SEVENTH *)
  END;  (* EVROOK *)



BEGIN
  IF XTMV[JNTM]*MBVAL[JNTK] + MAXPS <= BSTVL[JNTK-2] THEN
                                               (* MOVE WILL PRUNE ANYWAY *)
    INTV := XTMV[JNTM] * MBVAL[JNTK]
  ELSE
  BEGIN
    INTV :=(  FWPAWN*(EVPAWN(TPLOC[LP],S2,R2)-EVPAWN(TPLOC[DP],S4,R7))
            + FWMINM*(EVMOBL(LB,LN)            -EVMOBL(DB,DN)            )
            + FWMAJM*(EVMOBL(LR,LQ)            -EVMOBL(DR,DQ)            )
            + FWROOK*(EVROOK(TPLOC[LR],XRRS[R7])
                     -EVROOK(TPLOC[DR],XRRS[R2])                         )
            + FWKING*(EVKING(TPLOC[LK],TPLOC[LP])
                     -EVKING(TPLOC[DK],TPLOC[DP])                        )
           ) DIV 64;
    MAXPS := MAX(MAXPS,ABS(INTV));
    INTV := XTMV[JNTM]*(MBVAL[JNTK]+INTV);
  END;
  IF SWTR THEN
  BEGIN
    WRITE(" EVALU8",JNTK,JNTW,INDEX[JNTK],INTV);
    PRIMOV(MOVES[INDEX[JNTK]]);
  END;
  VALUE[INDEX[JNTK]] := INTV;                  (* RETURN SCORE *)
END;  (* EVALU8 *)


FUNCTION SEARCH                                (* SEARCH LOOK-AHEAD TREE *)
  :TW;                                         (* RETURNS THE BEST MOVE *)

LABEL
  11,                                          (* START NEW PLY *)
  12,                                          (* TRY DIFFERENT FIRST MOVE *)
  13,                                          (* FLOAT VALUE BACK UP *)
  14,                                          (* FIND ANOTHER MOVE *)
  15,                                          (* BACK UP A PLY *)
  16;                                          (* EXIT SEARCH *)


  PROCEDURE NEWBST                             (* SAVE BEST MOVE INFORMATION *)
    (A:TK);                                    (* PLY OF BEST MOVE *)

  VAR
    INTW : TW;                                 (* MOVES INDEX *)
    INRM : RM;                                 (* SCRATCH *)

  BEGIN
    BSTMV[A] := INDEX[A+1];                    (* SAVE BEST MOVE *)
    IF A = AK THEN                             (* AT FIRST PLY *)
    BEGIN
      INRM := MOVES[BSTMV[A]];                 (* SAVE BEST MOVE *)
      FOR INTW := BSTMV[A]-1 DOWNTO AW+1 DO
        MOVES[INTW+1] := MOVES[INTW];          (* MOVE OTHER MOVES DOWN *)
      MOVES[AW+1] := INRM;                     (* PUT BEST AT BEGINNING *)
      BSTMV[AK] := AW+1;                       (* POINTS TO BEST MOVE *)
    END
    ELSE
      IF NOT MOVES[BSTMV[A]].RMCA THEN
        KILLR[JNTK] := MOVES[BSTMV[A]];(* SAVE KILLER MOVE *)
  END;  (* NEWBST *)



  FUNCTION MINMAX                              (* PERFORM MINIMAX OPERATION *)
    (A:TK)                                     (* PLY TO MINIMAX AT *)
     :TB;                                      (* TRUE IF REFUTATION *)

  BEGIN
    MINMAX := FALSE;                           (* DEFAULT IS NO PRUNING *)
    IF SWTR THEN
      WRITE(" MINMAX",A,-BSTVL[A-1],BSTVL[A],-BSTVL[A+1]);
    IF -BSTVL[A+1] > BSTVL[A] THEN
    BEGIN
      BSTVL[A] := -BSTVL[A+1];
      NEWBST(A);                               (* SAVE BEST MOVE *)
      MINMAX := BSTVL[A+1] <= BSTVL[A-1];
                                               (* RETURN TRUE IF REFUTATION *)
      IF SWTR THEN
        WRITE(" NEW BEST. PRUNE: ",BSTVL[A+1] <= BSTVL[A-1]);
    END;
    IF SWTR THEN
      WRITELN;                                 (* PRINT TRACE LINE *)
  END;  (* MINMAX *)
```

```
  PROCEDURE SCOREM;                            (* SCORE MATE *)

  BEGIN
    MOVES[INDEX[JNTK]].RMMT := TRUE;     (* INDICATE MATE *)
    IF MOVES[INDEX[JNTK]].RMCH THEN      (* CHECKMATE *)
      VALUE[INDEX[JNTK]] :=  64*JNTK - ZV
    ELSE                                 (* STALEMATE *)
      VALUE[INDEX[JNTK]] := 0;
    IF SWTR THEN
      WRITELN(" SCOREM",JNTK,JNTW,INDEX[JNTK],VALUE[INDEX[JNTK]]);
  END;  (* SCOREM *)


  FUNCTION SELECT                              (* SELECT NEXT MOVE TO SEARCH *)
     :TB;                                      (* TRUE IF MOVE RETURNED *)

  LABEL
    21,                                        (* NEW SEARCH MODE *)
    22;                                        (* EXIT SELECT *)

  VAR
    INTB : TB;                                 (* RETURN VALUE *)
    INTK : TK;                                 (* SCRATCH *)
    INTW : TW;                                 (* MOVE INDEX *)
    INTM : TW;                                 (* SCRATCH *)
    INTV : TV;                                 (* SCRATCH *)


    PROCEDURE SELDON;                          (* SELECT EXIT - DONE.
                                                  CALLED WHEN NO FURTHER
                                                  MOVES ARE TO BE SEARCHED
                                                  FROM THIS POSITION.
                                                  THE CURRENT POSITION MUST
                                                  HAVE BEEN EVALUATED. *)

    BEGIN
      INTB := FALSE;                           (* RETURN NO MOVE SELECTED *)
      IF SWTR THEN
        WRITELN(" SELECT",JNTK," END.");
      GOTO 22;                                 (* EXIT SELECT *)
    END;  (* SELDON *)


    PROCEDURE SELMOV                           (* SELECT EXIT - SEARCH.
                                                  CALLED WHEN A MOVE TO
                                                  BE SEARCHED HAS BEEN
                                                  FOUND. *)
      (A:TW);                                  (* INDEX TO SELECTED MOVE *)

    BEGIN
      INTB := TRUE;                            (* RETURN MOVE SELECTED *)
      INDEX[JNTK+1] := A;                      (* POINT TO SELECTED MOVE *)
      MOVES[A].RMSU := TRUE;                   (* FLAG MOVE AS SEARCHED *)
      IF SWTR THEN
      BEGIN
        WRITE(" SELECT",JNTK,ORD(SRCHM[JNTK]),A);
        PRIMOV(MOVES[A]);
      END;
      GOTO 22;                                 (* EXIT SELECT *)
    END;  (* SELMOV *)


    PROCEDURE SELNXT                           (* SELECT EXIT - NEW MODE.
                                                  CALLED WHEN A NEW SEARCH
                                                  MODE IS TO BE SELECTED *)
      (A:TM);                                  (* NEW SEARCH MODE *)

    BEGIN
      INDEX[JNTK+1] := LINDX[JNTK]-1;          (* RESET MOVES POINTER *)
      SRCHM[JNTK] := A;                        (* CHANGE SEARCH MODE *)
      GOTO 21;                                 (* EXECUTE NEXT MODE *)
    END;  (* SELNXT *)


    PROCEDURE SELANY;                          (* SEARCH ALREADY GENERATED
                                                  AND NOT ALREADY SEARCHED *)

    VAR
      INTW : TW;                               (* MOVES INDEX *)

    BEGIN
      FOR INTW := INDEX[JNTK+1]+1 TO JNTW-1 DO
        IF NOT MOVES[INTW].RMSU THEN
          SELMOV(INTW);
    END;  (* SELANY *)


  BEGIN
  21:  (* NEW SEARCH MODE *)
    CASE SRCHM[JNTK] OF

      M0:  (* INITIALIZE FOR NEW MOVE *)
           BEGIN
             MVSEL[JNTK] := 0;               (* CLEAR MOVES SEARCHED *)
             INTV := BSTVL[JNTK-2];          (* SAVE ALPHA *)
             BSTVL[JNTK-2] := -ZV;           (* INHIBIT PRUNING IN EVALU8 *)
             MAXPS := 0;                     (* INITIALIZE MAXIMUM POSITIONAL
                                                SCORE *)
             GENALL;                         (* GENERATE ALL MOVES *)
             FOR INTW := AW+1 TO JNTW-1 DO
             BEGIN
               IF UPDATE(MOVES[INTW]) THEN
               BEGIN
                 INDEX[JNTK] := INTW;        (* POINT TO CURRENT MOVE *)
                 EVALU8;                     (* SCORE POSITION *)
               END;
               DNDATE(MOVES[INTW]);
             END;
             BSTVL[JNTK-2] := INTV;          (* RESTORE ALPHA *)
             SORTIT(VALUE,MOVES,JNTW-1);
                                             (* SORT PRELIMINARY SCORES *)
             FOR INTK := AK TO ZK DO
               KILLR[INTK] := NULMV;         (* CLEAR KILLER TABLE *)
```

Listing 1, continued:

```
      IF SWTR OR SWPS THEN
        FOR INTW := AW+1 TO JNTW-1 DO
        BEGIN
          WRITE(" PRELIM",INTW,VALUE[INTW]);
          PRIMOV(MOVES[INTW]);  (* PRINT PRELIMINARY SCORES *)
          IF INTW/LPP = INTW DIV LPP THEN
            PAUSER;
        END;
      SELNXT(H6);                  (* SEARCH ALL MOVES *)
    END;

H1:  (* INITIALIZE AT NEW DEPTH *)
    BEGIN
      MVSEL[JNTK] := 0;            (* CLEAR MOVES SEARCHED *)
      IF JNTK > JMTK THEN
      BEGIN
        EVALU8;                    (* EVALUATE CURRENT POSITION *)
        INDEX[JNTK+1] := AW;
        BSTVL[JNTK+1] := -VALUE[INDEX[JNTK]];
        IF MINMAX(JNTK) OR (JNTK = ZK) THEN
          SELDON;                  (* THIS MOVE PRUNES *)
        SRCHM[JNTK] := H2;         (* CAPTURE SEARCH *)
      END
      ELSE
        SRCHM[JNTK] := H3;         (* CAPTURES IN FULL SEARCH *)
      GENCAP;                      (* GENERATE CAPTURES *)
      SELNXT(SRCHM[JNTK]);         (* CHANGE SEARCH MODE *)
    END;

H2:  (* CAPTURE SEARCH *)
    BEGIN
      INTW := AW;                  (* BEST MOVE POINTER *)
      INTV := AV;                  (* BEST VALUE *)
      FOR INTW := LINDX[JNTK] TO JNTW-1 DO
        WITH MOVES[INTW] DO
          IF NOT RMSU THEN
            IF ABS(XTPV[RMCP]) > INTV THEN
            BEGIN
              INTV := ABS(XTPV[RMCP]);

              INTW := INTW;
            END;
      IF INTW <> AW THEN           (* MOVE FOUND *)
        SELMOV(INTW)               (* SELECT BIGGEST CAPTURE *)
      ELSE
        SELDON;                    (* QUIT *)
    END;

H3:  (* FULL WIDTH SEARCH - CAPTURES *)
    BEGIN
      INTW := AW;                  (* BEST MOVE POINTER *)
      INTV := AV;                  (* BEST VALUE *)
      FOR INTW := LINDX[JNTK] TO JNTW-1 DO
        WITH MOVES[INTW] DO
          IF NOT RMSU THEN
            IF ABS(XTPV[RMCP]) > INTV THEN
            BEGIN
              INTV := ABS(XTPV[RMCP]);

              INTW := INTW;
            END;
      IF INTW <> AW THEN           (* MOVE FOUND *)
        SELMOV(INTW)               (* SELECT BIGGEST CAPTURE *)
      ELSE
        IF NOT NULMVB(KILLR[JNTK]) THEN
        BEGIN
          INTW := JNTW;            (* SAVE CURRENT MOVES INDEX *)
          GENFSL(XRSS[KILLR[JNTK].RMFR]);
                                   (* GENERATE MOVE BY KILLER *)
          SRCHM[JNTK] := H4;       (* SET NEXT SEARCH MODE *)
          FOR INTW := INTW TO JNTW-1 DO
                                   (* LOOK AT MOVES BY KILLER *)
            IF KILLR[JNTK].RMTO = MOVES[INTW].RMTO THEN
              SELMOV(INTW);        (* SELECT KILLER MOVE *)
        END;
      SELNXT(H4);                  (* GO TO NEXT STATE *)
    END;

H4:  (* INITIALIZE SCAN OF CASTLE MOVES AND OTHER MOVES
       BY KILLER PIECE *)
    BEGIN
      GENCAS;                      (* GENERATE CASTLE MOVES *)
      SELNXT(H5);                  (* GO TO NEXT STATE *)
    END;

H5:  (* FULL WIDTH SEARCH - CASTLES AND OTHER MOVES BY KILLER
       PIECE *)
    BEGIN
      SELANY;                      (* SELECT ANY MOVE *)
      GENFSL(ALLOC[JNTK]);         (* GENERATE REMAINING MOVES *)
      SELNXT(H6);                  (* NEXT SEARCH MODE *)
    END;

H6:  (* FULL WIDTH SEARCH - REMAINING MOVES *)
    BEGIN
      SELANY;                      (* SELECT ANYTHING ON LIST *)
      IF MVSEL[JNTK] = 0 THEN
        SCOREM;                    (* SCORE MATE *)
      SELDON;                      (* EXIT SELECT *)
    END;

H7:  (* RESEARCH FIRST PLY *)
    BEGIN
      JNTW := LINDX[AK+1];         (* POINT TO ALREADY GENERATED
                                      MOVES *)
      MVSEL[AK] := 0;              (* RESET MOVES SEARCHED *)
      FOR INTW := AW+1 TO JNTW-1 DO
        MOVES[INTW].RMSU := FALSE;
                                   (* CLEAR SEARCHED BIT *)
      IF SWTR THEN
        WRITELN(" REDO  ",JNTK,BSTVL[AK-2],BSTVL[AK-1]);
      SELNXT(H6);                  (* SEARCH ALL MOVES *)
    END;
END;

  22:  (* SELECT EXIT *)
    SELECT := INTB;                         (* RETURN VALUE *)
  END;  (* SELECT *)

BEGIN  (* SEARCH *)
  BSTMV[AK] := AW;                          (* INITIALIZE MOVE *)
  INDEX[JNTK] := AW;                        (* INITIALIZE TREE *)
  MOVES[AW] := LSTMV;                       (* INITIALIZE MOVE *)
  EVALU8;                                   (* INITIAL GUESS AT SCORE *)
  BSTVL[AK-2] := VALUE[AW] - WINDOW;        (* INITIALIZE ALPHA-BETA
                                               WINDOW *)
  BSTVL[AK-1] := - VALUE[AW] - WINDOW;
  JNTK := AK+1;                             (* INITIALIZE ITERATION NUMBER *)
  WHILE (NODES < FNODEL) AND (JNTK < MAX(ZK DIV 2, ZK-8)) DO
  BEGIN

11:  (* START NEW PLY *)
    BSTVL[JNTK] := BSTVL[JNTK-2];           (* INITIALIZE ALPHA *)

12:  (* DIFFERENT FIRST MOVE *)
    IF NOT SELECT THEN
    BEGIN
      BSTVL[JNTK] := VALUE[INDEX[JNTK]];
      NEWBST(JNTK);
    END
    ELSE
    BEGIN
      IF UPDATE(MOVES[INDEX[JNTK+1]]) THEN
        GOTO 11                             (* START NEW PLY *)
      ELSE
      BEGIN
        DNDATE(MOVES[INDEX[JNTK]]);
        GOTO 12;                            (* FIND ANOTHER MOVE *)
      END;

13:  (* FLOAT VALUE BACK *)
      IF MINMAX(JNTK) THEN
        GOTO 15;                            (* PRUNE *)

14:  (* FIND ANOTHER MOVE AT THIS PLY *)
      IF SELECT THEN
        IF UPDATE(MOVES[INDEX[JNTK+1]]) THEN
          GOTO 11                           (* START NEW PLY *)
        ELSE
        BEGIN
          DNDATE(MOVES[INDEX[JNTK]]);
          GOTO 14;                          (* FIND ANOTHER MOVE *)
        END;
    END;

15:  (* BACK UP A PLY *)
    IF JNTK > AK THEN
    BEGIN (* NOT DONE WITH ITERATION *)
      DNDATE(MOVES[INDEX[JNTK]]);           (* RETRACT MOVE *)
      GOTO 13;
    END;

     (* DONE WITH ITERATION *)
    IF (BSTVL[AK] <= BSTVL[AK-2]) OR (BSTVL[AK] >= -BSTVL[AK-1]) THEN
    BEGIN  (* NO MOVE FOUND *)
      IF MVSEL[AK] = 0 THEN
      BEGIN  (* NO LEGAL MOVES *)
        GOTO 16;                            (* GIVE UP *)
      END;
      BSTVL[AK-2] := -ZV;                   (* SET ALPHA-BETA WINDOW LARGE *)
      BSTVL[AK-1] := -ZV;
      SRCHM[AK] := H7;
      JNTW := AK+1;
      GOTO 11;                              (* TRY AGAIN *)
    END;
    BSTVL[AK-2] := BSTVL[AK] - WINDOW;      (* SET ALPHA BETA WINDOW *)
    BSTVL[AK-1] := - BSTVL[AK] - WINDOW;
    JNTK := JNTK+1;                         (* ADVANCE ITERATION NUMBER *)
    SRCHM[AK] := H7;
  END;

16:  (* EXIT SEARCH *)
  SEARCH := BSTMV[AK];                      (* RETURN BEST MOVE *)
END;  (* SEARCH *)



PROCEDURE READER;                           (* READ INPUT FROM USER *)

LABEL
  11;                                       (* COMMAND FINISHED EXIT *)

VAR
  INRA : RA;                                (* SCRATCH TOKEN *)
  INTJ : TJ;                                (* ECHO COMMAND INDEX *)



  PROCEDURE RDRERR(A:RN);                   (* PRINT DIAGNOSTIC AND EXIT *)

  VAR
    INTJ : TJ;                              (* STRING INDEX *)
    INTN : TN;                              (* MESSAGE INDEX *)

  BEGIN
    IF NOT SWEC THEN                        (* ECHO LINE IF NOT ALREADY
                                               DONE *)
    BEGIN
      WRITE(" ");
      FOR INTJ := AJ TO ZJ-1 DO
        WRITE(ILINE[INTJ]);                 (* WRITE INPUT LINE *)
      WRITELN;
    END;
    FOR INTJ := AJ TO JNTJ DO
      WRITE(" ");                           (* LEADING BLANKS BEFORE ARROW *)
    WRITELN("^");                           (* POINTER TO ERROR *)
    FOR INTN := AN TO ZN DO
      WRITE(A[INTN]);                       (* WRITE DIAGNOSTIC *)
    WRITELN;
    GOTO 11;                                (* COMMAND EXIT *)
  END;  (* RDRERR *)
```

The LATEST in Tape Systems



NEW!

MODEL CC-9
$200.00 (4800 Baud)
$220.00 (9600 Baud and 220V/50 Hz)

9600 BAUD CASSETTE RECORDER

An ASYNCHRONOUS NRZ type Recorder with remote motor start/stop. Error rate 10^{8} at 4800 BAUD. Can be used from 110 to 9600 BAUD into a UART — no clocking required. This is not an audio recorder. It takes RS232 or TTL signals from the terminal or computer and gives back the same signals. No audio interface is used. Motor start/stop is manual or through TTL or RS232 signals.

Tape speeds are 1.6" / 3.0" and 6.0" per second. 110 volt, 60 Hz, 5 watts. (220 Volts on special order). Can use high quality audio cassettes (Philips Type) or certified data cassettes.

Recommended for DATA LOGGING, WORD PROCESSING, COMPUTER PROGRAM RELOADING and DATA STORAGE. Manual control except for motor start/stop. 6800, 8080 or Z80 software for file or record searching available on request with order. Used by major computer manufacturers, Bell Telephone and U.S. Government for program reloading and field servicing.

AVAILABILITY — Off the shelf.

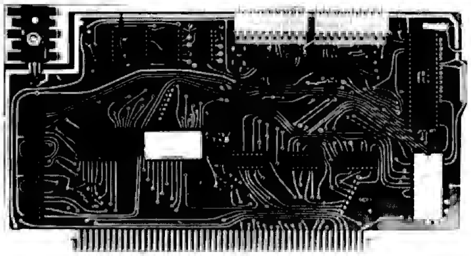
PROVIDES MONITOR AND TAPE SOFTWARE in ROM. TERMINAL and TAPE PORTS on SAME BOARD. CONTROLS ONE or TWO TAPE UNITS (CC-8 or 3M3B).

This is a complete 8080, 8085, or Z80 system controller. It provides the terminal I/O (RS232, 20 mA or TTL) and the data cartridge I/O, plus the motor controlling parallel I/O latches. Two kilobytes of on board ROM provide turn on and go control of your Altair or IMSAI. NO MORE BOOTSTRAPPING. Loads and Dumps memory in hex on the terminal, formats tape cartridge files, has word processing and paper tape routines. Best of all, it has the search routines to locate files and records by means of six, five, and four letter strings. Just type in the file name and the recorder and software do the rest. Can be used in the BiSync (IBM), BiPhase (Phase encoded) or NRZ modes with suitable recorders, interfaces and software.

This is Revision 8 of this controller. This version features 2708 type EPROM's so that you can write your own software or relocate it as desired. One 2708 preprogrammed is supplied with the board. A socket is available for the second ROM allowing up to a full 2K of monitor programs.

Fits all S100 bus computers using 8080 or Z80 MPU's. Requires 2 MHz clock from bus. Cannot be used with audio cassettes without an interface. Cassette or cartridge inputs are TTL or RS232 level.

AVAILABILITY — Off the shelf.



2SIO (R) CONTROLLER
$190.00, Tested & Assmb.

NEW!

→ DOUBLE DENSITY FLOPPY DISK CONTROLLER

A new floppy controller for 5" and 8" drives utilizing the new 1791 chip to provide single or double density recording. Flip the switch to use one or the other mode. Can load memory from single density and re-record it double density on the same drive so you can transfer or re-record your programs and files. Comes with new format program for double density on disk to replace your old single density format program. (Soft Sector IBM format). $295. Assembled and tested.

SHIPPING STARTED OCTOBER '78.

NEW!

→ PEGASUS DOUBLE DISK ENCLOSURE and POWER SUPPLY

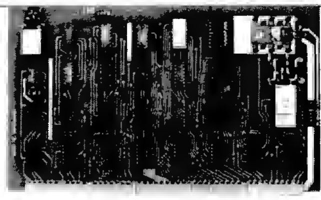
Holds one or two drives, has two drive power supply. 110 or 220V. Comes empty or filled with Shugart drives. Variations preclude listing prices. Call or write for full details.

SHIPPING STARTED OCTOBER '78.

Z 80 BOARD for SWTP COMPUTER

Now you can use the 8080/Z80 software programs in your SWTP 6800 machine. Replaces your MPU board with a Z80 and ROM so that you are up and running with your present SWTP memory and MPS card. 1 K ROM on board replaces MIKBUG and enables you to use XITAN Z80 software which we can supply.

AVAILABILITY — Off the shelf.



$190.00, Tested & Assmb.

For U.P.S. delivery, add $3.00. Overseas and air shipments charges collect, N.J. Residents add 5% Sales Tax. WRITE or CALL for further information. Phone Orders on Master Charge and BankAmericard accepted.

National Multiplex Corporation

3474 Rand Avenue, South Plainfield NJ 07080 Box 288 Phone (201) 561-3600 TWX 710-997-9530

Listing 1, continued:

```
FUNCTION RDRGNT(VAR A:RA):TB;       (* GET NEXT TOKEN FROM COMMAND
                                       RETURNS TOKEN IN A.
                                       RETURNS TRUE IF NON-EMPTY
                                       TOKEN.
                                       A TOKEN IS ANY CONSECUTIVE
                                       COLLECTION OF ALPHANUMERIC
                                       CHARACTERS.
                                       LEADING SPECIAL CHARACTERS
                                       IGNORED. *)


VAR
  INTJ : TJ;                        (* STRING INDEX *)

BEGIN
  WHILE (JNTJ < ZJ) AND (ORD(ILINE[JNTJ]) >= ORD("+")) DO
    JNTJ := JNTJ+1;
  A := "          ";
  INTJ := AA;
  WHILE (JNTJ < ZJ) AND (INTJ < ZA) AND (ILINE[JNTJ] IN ["A".."9"]) DO
  BEGIN
    A[INTJ] := ILINE[JNTJ];         (* COPY CHARACTER TO TOKEN *)
    INTJ := INTJ+1;                 (* ADVANCE POINTERS *)
    JNTJ := JNTJ+1;
  END;
  RDRGNT := INTJ <> AA;             (* RETURN TRUE IF ANYTHING
                                       MOVED *)
  WHILE (INTJ < ZJ) AND (ILINE[JNTJ] IN ["A".."9"]) DO
    JNTJ := JNTJ+1;                 (* SKIP REST OF TOKEN *)
END;  (* RDRGNT *)



PROCEDURE RDRSFT;                   (* SKIP FIRST TOKEN IN COMMAND
                                       LINE *)

VAR
  INRA : RA;                        (* SCRATCH *)
  INTB : TB;                        (* SCRATCH *)

BEGIN
  JNTJ := AJ;                       (* INITIALIZE SCAN *)
  INTB := RDRGNT(INRA);             (* THROW AWAY FIRST TOKEN *)
END;  (* RDRSFT *)



PROCEDURE RDRCMD                    (* TEST FOR AND EXECUTE COMMAND
                                       EXITS TO COMMAND EXIT IF
                                       COMMAND IS PROCESSED. *)
```



```
  (A:RA;                            (* POTENTIAL COMMAND KEYWORD *)
   PROCEDURE XXXCMD);               (* PROCEDURE TO EXECUTE
                                       COMMAND *)

BEGIN
  IF INRA = A THEN
  BEGIN
    XXXCMD;                         (* EXECUTE COMMAND *)
    GOTO 11;                        (* EXIT *)
  END;
END;  (* RDRCMD *)


PROCEDURE RDLINE;                   (* GET NEXT INPUT LINE FROM
                                       USER *)

VAR
  INTC : TC;                        (* SCRATCH *)
  INTJ : TJ;                        (* STRING INDEX *)

BEGIN
  READLN;                           (* ADVANCE TO NEXT LINE *)
  INTJ := AJ;
  WHILE NOT EOLN AND (INTJ < ZJ) DO
  BEGIN
    READ(ICARD[INTJ]);              (* COPY INPUT LINE *)
    INTJ := INTJ+1;
  END;
  WHILE NOT EOLN DO
    READ(INTC);                     (* SKIP REST OF INPUT LINE *)
  WHILE INTJ < ZJ DO
  BEGIN
    ICARD[INTJ] := " ";             (* BLANK REST OF LINE *)
    INTJ := INTJ+1;
  END;
  ICARD[ZJ] := ";";                 (* SET END OF COMMAND *)
  JNTJ := AJ;                       (* RESET INPUT LINE POINTER *)
END;  (* RDLINE *)


FUNCTION RDRMOV:TB;                 (* EXTRACT NEXT COMMAND
                                       FROM INPUT LINE.
                                       RETURNS TRUE IF NON-EMPTY
                                       COMMAND. *)

VAR
  INTJ : TJ;                        (* STORING POINTER *)

BEGIN
  WHILE (JNTJ < ZJ) AND (ICARD[JNTJ] = " ") DO
    JNTJ := JNTJ+1;                 (* SKIP LEADING BLANKS *)
  INTJ := AJ;
  WHILE (JNTJ < ZJ) AND (ICARD[JNTJ] <> ";") DO
  BEGIN
    ILINE[INTJ] := ICARD[JNTJ];
    INTJ := INTJ+1;
    JNTJ := JNTJ+1;
  END;
  IF (ICARD[JNTJ] = ";") AND (JNTJ < ZJ) THEN
    JNTJ := JNTJ+1;                 (* SKIP SEMI-COLON *)
  RDRMOV := INTJ <> AJ;             (* RETURN TRUE IF NON-EMPTY *)
  WHILE INTJ < ZJ DO
  BEGIN
    ILINE[INTJ] := " ";             (* BLANK FILL LINE *)
    INTJ := INTJ+1;
  END;
  ILINE[ZJ] := ";";                 (* STORE COMMAND TERMINATOR *)
  JNTJ := AJ;                       (* PRESET COMMAND SCAN *)
END;  (* RDRMOV *)


FUNCTION RDRNUM:TI;                 (* CRACK NUMBER FROM COMMAND
                                       LINE.  RETURNS NUMBER IF NO
                                       ERROR.  EXITS TO COMMAND EXIT
                                       IF ERROR. *)

VAR
  INTB : TB;                        (* SIGN *)
  INTI : TI;                        (* VALUE *)

BEGIN
  WHILE (JNTJ < ZJ) AND (ILINE[JNTJ] = " ") DO
    JNTJ := JNTJ+1;                 (* SKIP LEADING BLANKS *)
  IF ILINE[JNTJ] = "-" THEN
  BEGIN
    INTB := TRUE;                   (* NUMBER IS NEGATIVE *)
    JNTJ := JNTJ+1;                 (* ADVANCE CHARACTER POINTER *)
  END
  ELSE
  BEGIN
    INTB := FALSE;                  (* NUMBER IS POSITIVE *)
    IF ILINE[JNTJ] = "+" THEN
      JNTJ := JNTJ+1;               (* SKIP LEADING + *)
  END;
  INTI := 0;
  WHILE ILINE[JNTJ] IN ["0".."9"] DO
  BEGIN
    IF INTI < MAXINT/10 THEN
      INTI := 10*INTI+ORD(ILINE[JNTJ])-ORD("0")
    ELSE
      RDRERR(" NUMBER TOO LARGE            ");
    JNTJ := JNTJ+1;                 (* ADVANCE *)
  END;
  IF ILINE[JNTJ] IN ["A".."Z"] THEN
    RDRERR(" DIGIT EXPECTED              ");
  IF INTB THEN
    INTI := -INTI;                  (* COMPLEMENT IF NEGATIVE *)
  RDRNUM := INTI;                   (* RETURN NUMBER *)
END;  (* RDRNUM *)


PROCEDURE BOACMD;                   (* COMMAND - SET UP POSITION *)

VAR
  INTM : TM;                        (* COLOR *)
  INTS : TS;                        (* POSITION ON BOARD *)
```

Listing 1, continued:

```
  PROCEDURE BOAADV(A:TI);              (* ADVANCE N FILES *)

  BEGIN
    IF INTS+A < ZS THEN
      INTS := INTS+A
    ELSE
      INTS := ZS;
  END;  (* BOAADV *)


  PROCEDURE BOASTO(A:TP);              (* STORE PIECE ON BOARD *)

  BEGIN
    BOARD.RBIS[INTS] := A;
    IF INTS < ZS THEN
      INTS := INTS+1;
  END;  (* BOASTO *)


BEGIN  (* BOACMD *)
  CLSTAT;                              (* CLEAR STATUS FLAGS *)
  LSTMV := NULMV;                      (* CLEAR PREVIOUS MOVE *)
  FOR INTS := AS TO ZS DO
    BOARD.RBIS[INTS] := MT;            (* CLEAR BOARD *)
  INTM := LITE;
  INTS := 8;
  REPEAT
    IF ILINE[JNTJ] IN ["P","R","N","B","Q","K","L","D","1".."8"] THEN
    CASE ILINE[JNTJ] OF
      "P": BOASTO(XTUMP[EP,INTM]);
      "R": BOASTO(XTUMP[ER,INTM]);
      "N": BOASTO(XTUMP[EN,INTM]);
      "B": BOASTO(XTUMP[EB,INTM]);
      "Q": BOASTO(XTUMP[EQ,INTM]);
      "K": BOASTO(XTUMP[EK,INTM]);
      "L": INTM := LITE;
      "D": INTM := DARK;
      "1","2","3","4","5","6","7","8":
           BOAADV(ORD(ILINE[JNTJ])-ORD("0"));
    END
    ELSE
      IF ILINE[JNTJ] IN ["A".."9"] THEN
      BEGIN
        FOR INTS := AS TO ZS DO
          BOARD.RBIS[INTS] := MT;
        CLSTAT;                        (* CLEAR STATUS *)
        RORERR(" ILLEGAL BOARD OPTION          ");
      END;
    JNTJ := JNTJ+1;
  UNTIL JNTJ = ZJ;
END;  (* BOACMD *)


PROCEDURE ENDCMD;                      (* COMMAND - END PROGRAM *)

BEGIN
  GOTO 9;                              (* END PROGRAM *)
END;  (* ENDCMD *)


PROCEDURE GONCMD;                      (* COMMAND - GO N MOVES *)

BEGIN
  GOING := RORNUM;                     (* CRACK NUMBER *)
  IF GOING <= 0 THEN
    GOING := 1;
  GOTO 2;                              (* EXECUTE MACHINES MOVE *)
END;  (* GONCMD *)


PROCEDURE INICMD;                      (* COMMAND - INITIALIZE FOR A NEW
                                          GAME *)

BEGIN
  GOTO 1;                              (* INITIALIZE FOR A NEW GAME *)
END;  (* INICMD *)


PROCEDURE LETCMD;                      (* COMMAND - CHANGE VARIABLE *)

LABEL
  21;                                  (* LET COMMAND EXIT *)

PROCEDURE LETONE                       (* TEST FOR AND SET ONE
                                          VARIABLE *)
  (A:RA;                               (* VARIABLE NAME *)
   VAR B:TI);                          (* VARIABLE *)

BEGIN
  IF A = INRA THEN
  BEGIN
    B := RORNUM;                       (* GET VALUE *)
    GOTO 21;                           (* EXIT *)
  END;
END;  (* LETONE *)


BEGIN
  IF RORGNT(INRA) THEN
  BEGIN
    LETONE("FKPSHD    ",FKPSHD);
    LETONE("FKSANQ    ",FKSANQ);
    LETONE("FMAXMT    ",FMAXMT);
    LETONE("FNODEL    ",FNODEL);
    LETONE("FPADQR    ",FPADCR[F1]);
    LETONE("FPADQN    ",FPADCR[F2]);
    LETONE("FPADQB    ",FPADCR[F3]);
    LETONE("FPADQF    ",FPADCR[F4]);
    LETONE("FPADKF    ",FPADCR[F5]);
    LETONE("FPADKB    ",FPADCR[F6]);
    LETONE("FPADKN    ",FPADCR[F7]);
    LETONE("FPADKR    ",FPADCR[F8]);
    LETONE("FPBLOK    ",FPBLOK);
    LETONE("FPCONN    ",FPCONN);
    LETONE("FPFLNX    ",FPFLNX);
    LETONE("FRDUBL    ",FRDUBL);
    LETONE("FRK7TH    ",FRK7TH);
    LETONE("FTRADE    ",FTRADE);
    LETONE("FTRDSL    ",FTRDSL);
    LETONE("FTRPOK    ",FTRPOK);
    LETONE("FTRPWN    ",FTRPWN);
    LETONE("FWKING    ",FWKING);
    LETONE("FWMAJM    ",FWMAJM);
    LETONE("FWMINM    ",FWMINM);
    LETONE("FWPAWN    ",FWPAWN);
    LETONE("FWROOK    ",FWROOK);
    LETONE("WINDOW    ",WINDOW);
    RORERR(" ILLEGAL LET VARIABLE NAME    ");
  END;
21:  (* LET COMMAND EXIT *)
END;  (* LETCMD *)


PROCEDURE PLECMD;                      (* COMMAND - PRINT VARIABLE *)

LABEL
  21;                                  (* PRINT LET COMMAND EXIT *)

PROCEDURE PRIONE                       (* TEST FOR AND PRINT VARIABLE *)
  (A:RA;                               (* TEST VARIABLE NAME *)
   B:TI);                              (* VARIABLE *)

BEGIN
  IF INRA = A THEN
  BEGIN
    WRITELN(A,B);
    GOTO 21;                           (* EXIT *)
  END;
END;  (* PRIONE *)


BEGIN  (* PLECMD *)
  WHILE RORGNT(INRA) DO
  BEGIN
    PRIONE("FKPSHD    ",FKPSHD);
    PRIONE("FKSANQ    ",FKSANQ);
    PRIONE("FMAXMT    ",FMAXMT);
    PRIONE("FNODEL    ",FNODEL);
    PRIONE("FPADQR    ",FPADCR[F1]);
    PRIONE("FPADQN    ",FPADCR[F2]);
    PRIONE("FPADQB    ",FPADCR[F3]);
    PRIONE("FPADQF    ",FPADCR[F4]);
    PRIONE("FPADKF    ",FPADCR[F5]);
    PRIONE("FPADKB    ",FPADCR[F6]);
    PRIONE("FPADKN    ",FPADCR[F7]);
    PRIONE("FPADKR    ",FPADCR[F8]);
    PRIONE("FPBLOK    ",FPBLOK);
    PRIONE("FPCONN    ",FPCONN);
    PRIONE("FPFLNX    ",FPFLNX);
    PRIONE("FRDUBL    ",FRDUBL);
    PRIONE("FRK7TH    ",FRK7TH);
    PRIONE("FTRADE    ",FTRADE);
    PRIONE("FTRDSL    ",FTRDSL);
    PRIONE("FTRPOK    ",FTRPOK);
    PRIONE("FTRPWN    ",FTRPWN);
    PRIONE("FWKING    ",FWKING);
    PRIONE("FWMAJM    ",FWMAJM);
    PRIONE("FWMINM    ",FWMINM);
    PRIONE("FWPAWN    ",FWPAWN);
    PRIONE("FWROOK    ",FWROOK);
    PRIONE("WINDOW    ",WINDOW);
    RORERR(" ILLEGAL VARIABLE NAME        ");

21:  (* PRINT LET CCMMAND EXIT *)
  END;
END;  (* PLECMD *)


PROCEDURE PRICMD;                      (* COMMAND - PRINT BOARD *)

BEGIN
  IF RORGNT(INRA) THEN
    PRINTB(NBORD)
  ELSE
    PRINTB(BOARD.RBIS);
END;  (* PRICMD *)


PROCEDURE PAMCMD;                      (* COMMAND - PRINT ATTACK MAP *)

BEGIN
  WHILE RORGNT(INRA) DO
    IF INRA[AA] = "T" THEN
      PRINAM(ATKTO)
    ELSE
      IF INRA[AA] = "F" THEN
        PRINAM(ATKFR)
      ELSE
        RORERR(" ATTACK MAP NOT 'TO' OR 'FROM'");
END;  (* PAMCMD *)


PROCEDURE POPCMD;                      (* COMMAND - PRINT OTHER STUFF *)

VAR
  INTQ : TQ;                           (* CASTLE TYPE INDEX *)

BEGIN
  WITH BOARD DO
  BEGIN
    WRITELN(XTMA[RBTM]," TO MOVE.");
```

Listing 1, continued:

```
    WRITELN(RBTS," ENPASSANT.");
    WRITELN("MOVE NUMBER",RBTI);
    FOR INTQ := LS TO DL DO
      IF INTQ IN RBSQ THEN
        WRITELN(XTQA[INTQ]," SIDE CASTLE LEGAL.");
  END;
END;  (* POPCMD *)


PROCEDURE PMVCMD;                      (* COMMAND - PRINT MOVE LIST *)

VAR
  INTW : TW;                           (* MOVES LIST INDEX *)

BEGIN
  LSTMOV;                              (* LIST LEGAL MOVES *)
  FOR INTW := AW TO JNTW-1 DO
  BEGIN
    WRITE(INTW:4," ");
    PRIMOV(MOVES[INTW]);
    IF INTW/LPP = INTW DIV LPP THEN
      PAUSER;
  END;
END;  (* PMVCMD *)


PROCEDURE SWICMD;                      (* COMMAND - FLIP SWITCH *)

LABEL
  21;                                  (* SWITCH OPTION EXIT *)


  PROCEDURE SWIONE                     (* PROCESS ONE SWITCH *)
    (A:RA;                             (* SWITCH NAME *)
     VAR B:TB);                        (* SWITCH *)

  VAR
    INTJ : TJ;                         (* SAVE COMMAND INDEX *)

  BEGIN
    IF INRA = A THEN
    BEGIN
      INTJ := JNTJ;                    (* SAVE CURRENT POSITION *)
      IF RDRGNT(INRA) THEN
      BEGIN
        IF INRA = "ON        " THEN
          B := TRUE                    (* TURN SWITCH ON *)
        ELSE
          IF INRA = "OFF       " THEN
            B := FALSE                 (* TURN SWITCH OFF *)
          ELSE
            JNTJ := INTJ;              (* RESTORE CURRENT POSITION *)
        PRISWI(A,B);                   (* PRINT SWITCH VALUE *)
      END
      ELSE
        PRISWI(A,B);
      GOTO 21;                         (* SWITCH OPTION EXIT *)
    END;
  END;  (* SWIONE *)


BEGIN  (* SWICMD *)
21:  (* SWITCH OPTION EXIT *)
  WHILE RDRGNT(INRA) DO
  BEGIN
    SWIONE("EC        ",SWEC);
    SWIONE("PA        ",SWPA);
    SWIONE("PS        ",SWPS);
    SWIONE("RE        ",SWRE);
    SWIONE("SU        ",SWSU);
    SWIONE("TR        ",SWTR);
    RDRERR(" INVALID SWITCH OPTION         ");
  END;
END;  (* SWICMD *)


PROCEDURE STACMD;                      (* COMMAND - STATUS CHANGES *)

LABEL
  21;                                  (* STATUS COMMAND OPTION EXIT *)
VAR
  INRA : RA;                           (* CURRENT TOKEN *)
  INTM : TM;                           (* SIDE BEING PROCESSED *)


  PROCEDURE STAEPF                     (* PROCESS EP FILE *)
    (A:RA;                             (* TEST TOKEN *)
     B:TF);                            (* EQUIVALENT FILE *)

  BEGIN
    IF A = INRA THEN
    BEGIN
      IF INTM = LITE THEN
        BOARD.RBTS := XTRFS[R6,B]
      ELSE
        BOARD.RBTS := XTRFS[R3,B];
      GOTO 21;                         (* EXIT STATUS OPTION *)
    END;
  END;  (* STAEPF *)


  PROCEDURE STACAK;                    (* ALLOW CASTLE KING SIDE *)

  BEGIN
    IF INTM = LITE THEN
      BOARD.RBSQ := BOARD.RBSQ + [LS]
    ELSE
      BOARD.RBSQ := BOARD.RBSQ + [DS];
  END;  (* STACAK *)


  PROCEDURE STACAQ;                    (* ALLOW CASTLE QUEEN SIDE *)

  BEGIN
    IF INTM = LITE THEN
      BOARD.RBSQ := BOARD.RBSQ + [LL]
    ELSE
      BOARD.RBSQ := BOARD.RBSQ + [DL];
  END;  (* STACAQ *)


  PROCEDURE STADRK;                    (* SET BLACK OPTIONS *)

  BEGIN
    INTM := DARK;
  END;  (* STADRK *)


  PROCEDURE STAENP;                    (* SET ENPASSANT FILE *)

  BEGIN
    IF NOT RDRGNT(INRA) THEN
    BEGIN
      CLSTAT;                          (* CLEAR STATUS *)
      RDRERR(" ENPASSANT FILE OMITTED       ");
    END;

    STAEPF("QR        ",F1);
    STAEPF("QN        ",F2);
    STAEPF("QB        ",F3);
    STAEPF("Q         ",F4);
    STAEPF("K         ",F5);
    STAEPF("KB        ",F6);
    STAEPF("KN        ",F7);
    STAEPF("KR        ",F8);
    CLSTAT;                            (* CLEAR STATUS *)
    RDRERR(" ILLEGAL ENPASSANT FILE       ");
  END;  (* STAENP *)


  PROCEDURE STAGOS;                    (* SET SIDE TO MOVE *)

  BEGIN
    BOARD.RBTM := INTM;
    JNTM := INTM;
  END;  (* STAGOS *)


  PROCEDURE STALIT;                    (* SET WHITE OPTIONS *)

  BEGIN
    INTM := LITE;
  END;  (* STALIT *)


  PROCEDURE STANUM;                    (* SET MOVE NUMBER *)

  BEGIN
    BOARD.RBTI := RDRNUM;
  END;  (* STANUM *)


  PROCEDURE STAOPT                     (* TEST STATUS OPTION *)
    (A:RA;                             (* TEST OPTION *)
     PROCEDURE STAXXX);                (* PROCEDURE TO EXECUTE IF
                                          EQUAL *)

  BEGIN
    IF INRA = A THEN
    BEGIN
      STAXXX;                          (* EXECUTE PROCEDURE *)
      GOTO 21;                         (* EXIT STATUS OPTION *)
    END;
  END;  (* STAOPT *)


BEGIN  (* STACMD *)
  CLSTAT;                              (* CLEAR STATUS *)
  INTM := LITE;                        (* DEFAULT SIDE WHITE *)
21:  (* STATUS OPTION EXIT *)
  WHILE RDRGNT(INRA) DO
  BEGIN
    STAOPT("D         ",STADRK);
    STAOPT("EP        ",STAENP);
    STAOPT("G         ",STAGOS);
    STAOPT("L         ",STALIT);
    STAOPT("N         ",STANUM);
    STAOPT("OO        ",STACAK);
    STAOPT("OOO       ",STACAQ);
    CLSTAT;
    RDRERR(" INVALID STATUS OPTION         ");
  END;
END;  (* STACMD *)


PROCEDURE WHACMD;                      (* COMMAND - WHAT? *)

BEGIN
  WRITELN(MOVMS);                      (* PRINT LAST MESSAGE *)
END;  (* WHACMD *)


BEGIN  (* READER *)
11: (* COMMAND EXIT *)
  WHILE NOT RDRMOV DO
    RDLINE;
```

Listing 1, continued:

```
  IF SWEC THEN
  BEGIN                                  (* ECHO LINE *)
    WRITE(" ");
    FOR INTJ := AJ TO ZJ-1 DO
      WRITE(ILINE[INTJ]);
    WRITELN;
  END;
  IF ILINE[AJ+1] IN ["A".."W","Y","Z"] THEN
  BEGIN
    INRA := "          ";                (* EXTRACT KEYWORD *)
    INRA[AA] := ILINE[AJ];
    INRA[AA+1] := ILINE[AJ+1];
    RDRSFT;                              (* SKIP FIRST TOKEN *)
    RDRCMD("BO        ",BOACMD);
    RDRCMD("EN        ",ENDCMD);
    RDRCMD("GO        ",GONCMD);
    RDRCMD("IN        ",INICMD);
    RDRCMD("LE        ",LETCMD);
    RDRCMD("PB        ",PAMCMD);
    RDRCMD("PO        ",POPCMD);
    RDRCMD("PL        ",PLECMD);
    RDRCMD("PM        ",PMVCMD);
    RDRCMD("PR        ",PRICMD);
    RDRCMD("ST        ",STACMD);
    RDRCMD("SW        ",SWICMD);
    RDRCMD("WH        ",WHACMD);
    RDRERR(" INVALID COMMAND               ");
  END;
END;  (* READER *)


PROCEDURE MINENG                         (* GENERATE MINIMUM
                                            ENGLISH NOTATION *)
  (A:RM;                                 (* MOVE TO NOTATE *)
   B:RA);                                (* LEADING COMMENT *)

VAR
  INTN : TN;                             (* MESSAGE INDEX *)

  PROCEDURE ADDCHR                       (* ADD CHARACTER TO MESSAGE *)
    (A:TC);                              (* CHARACTER *)

  BEGIN
    MOVMS[INTN] := A;                    (* ADD CHARACTER *)
    IF INTN < ZN THEN
      INTN := INTN+1;                    (* ADVANCE POINTER *)
  END;  (* ADDCHR *)


  PROCEDURE ADDSQR                       (* ADD SQUARE TO MESSAGE *)
    (A:TS;                               (* SQUARE TO ADD *)
     B:RD);                              (* SQUARE SYNTAX *)

  BEGIN
    WITH B DO
    BEGIN
      IF RDPC THEN
        ADDCHR(XTUC[XTPU[NBORD[A]]]);
      IF RDSL THEN
        ADDCHR("/");
      IF RDKQ THEN
        IF XTSF[A] IN [F1..F4] THEN
          ADDCHR("Q")
        ELSE
          ADDCHR("K");
      IF RDNB THEN
        CASE XTSF[A] OF
          F1,F8: ADDCHR("R");
          F2,F7: ADDCHR("N");
          F3,F6: ADDCHR("B");
          F4   : ADDCHR("Q");
          F5   : ADDCHR("K");
        END;
      IF RDRK THEN
        IF JNTM = LITE THEN
          CASE XTSR[A] OF
            R1: ADDCHR("1");
            R2: ADDCHR("2");
            F3: ADDCHR("3");
            R4: ADDCHR("4");
            R5: ADDCHR("5");
            R6: ADDCHR("6");
            R7: ADDCHR("7");
            R8: ADDCHR("8");
          END
        ELSE
          CASE XTSR[A] OF
            R1: ADDCHR("8");
            R2: ADDCHR("7");
            R3: ADDCHR("6");
            R4: ADDCHR("5");
            R5: ADDCHR("4");
            R6: ADDCHR("3");
            R7: ADDCHR("2");
            R8: ADDCHR("1");
          END;
    END;
  END;  (* ADDSQR *)


  PROCEDURE ADDWRD                       (* ADD WORD TO MESSAGE *)
    (A:RA;                               (* TEXT OF WORD *)
     B:TA);                              (* LENGTH OF WORD *)

  VAR
    INTA : TA;                           (* CHARACTER INDEX *)

  BEGIN
    FOR INTA := AA TO B DO
      ADDCHR(A[INTA]);
  END;  (* ADDWRD *)

  FUNCTION DIFFER                        (* COMPARE MOVES *)
    (A,B:RM)                             (* MOVES TO COMPARE *)
     :TB;                                (* TRUE IF MOVES ARE DIFFERENT *)

  VAR
    INTB : TB;                           (* SCRATCH *)

  BEGIN
    INTB := (A.RMFR <> B.RMFR) OR
            (A.RMTO <> B.RMTO) OR
            (A.RMCP <> B.RMCP);
    IF A.RMPR = B.RMPR THEN
      IF A.RMPR THEN
        DIFFER := INTB OR (A.RMPP <> B.RMPP)
      ELSE
        IF A.RMOO = B.RMOO THEN
          IF A.RMOO THEN
            DIFFER := INTB OR (A.RMQS <> B.RMQS)
          ELSE
            DIFFER := INTB
        ELSE
          DIFFER := TRUE
    ELSE
      DIFFER := TRUE;
  END;  (* DIFFER *)


  PROCEDURE SETSQD                       (* DEFINE SPECIFIC SQUARE
                                            DESCRIPTOR *)
    (A:TS;                               (* SQUARE TO DESCRIBE *)
     B:RD;                               (* SYNTAX TO USE *)
     VAR C:SR;                           (* SET OF POSSIBLE RANKS *)
     VAR D:SF);                          (* SET OF POSSIBLE FILES *)

  BEGIN
    C := [R1..R8];                       (* INITIALIZE TO DEFAULTS *)
    D := [F1..F8];
    WITH B DO
    BEGIN
      IF RDKQ AND RDNB THEN
        D := [XTSF[A]];
      IF (NOT RDKQ) AND RDNB THEN
        CASE XTSF[A] OF
          F1,F8: D := [F1,F8];
          F2,F7: D := [F2,F7];
          F3,F6: D := [F3,F6];
          F4   : D := [F4];
          F5   : D := [F5];
        END;
      IF RDRK THEN
        C := [XTSR[A]];
    END;
  END;  (* SETSQD *)


  PROCEDURE MINGEN                       (* PRODUCE MINIMUM
                                            ENGLISH NOTATION FOR
                                            MOVES AND CAPTURES *)
    (A:RM;                               (* MOVE OR CAPTURE *)
     B:TI;                               (* FIRST SYNTAX TABLE ENTRY *)
     C:TI);                              (* LAST SYNTAX TABLE ENTRY *)

  LABEL
    21,                                  (* EXIT AMBIGUOUS MOVE SCAN *)
    22;                                  (* EXIT MINGEN *)

  VAR
    INTG : TG;                           (* PROMOTION PIECE *)
    INTI : TI;                           (* SYNTAX TABLE INDEX *)
    INTW : TW;                           (* MOVES INDEX *)
    INLR : SR;                           (* RANKS DEFINED ON LEFT *)
    INRR : SR;                           (* RANKS DEFINED ON RIGHT *)
    INLF : SF;                           (* FILES DEFINED ON LEFT *)
    INRF : SF;                           (* FILES DEFINED ON RIGHT *)

  BEGIN
    FOR INTI := B TO C DO                (* FOR EACH SYNTAX ENTRY *)
      WITH SYNTX[INTI] DO
      BEGIN
        IF A.RMPR THEN
          INTG := A.RMPP
        ELSE
          INTG := PB;
        SETSQD(A.RMFR,RYLS,INLR,INLF);  (* SET SQUARE SETS *)
        SETSQD(A.RMTO,RYRS,INRR,INRF);
        FOR INTW := AW+1 TO JNTW-1 DO
          IF DIFFER(MOVES[INTW],A) THEN
            IF (NBORD[A.RMFR] = NBORD[MOVES[INTW].RMFR]) AND
               (A.RMCP = MOVES[INTW].RMCP) THEN
              WITH MOVES[INTW] DO
                IF (XTSR[RMFR] IN INLR) AND
                   (XTSR[RMTO] IN INRR) AND
                   (XTSF[RMFR] IN INLF) AND
                   (XTSF[RMTO] IN INRF) AND
                  ((RMPR AND (INTG = RMPP)) OR (NOT RMPR)) THEN
                  GOTO 21;               (* ANOTHER MOVE LOOKS THE SAME *)

        (* NO OTHER MOVE LOOKS THE SAME *)
        ADDSQR(A.RMFR,RYLS);            (* ADD FROM SQUARE *)
        ADDCHR(RYCH);                   (* ADD MOVE OR CAPTURE *)
        ADDSQR(A.RMTO,RYRS);            (* ADD TO SQUARE *)
        GOTO 22;                        (* EXIT MINGEN *)
    21:  (* TRY NEXT SYNTAX *)
      END;
    22:  (* EXIT MINGEN *)
  END;  (* MINGEN *)


BEGIN  (* MINENG *)
  MOVMS := "                             ";
                                        (* CLEAR MESSAGE *)
  INTN := AN+1;                         (* INITIALIZE MESSAGE INDEX *)
  ADDWRD(B,ZA);                         (* ADD INITIAL COMMENT *)
  ADDWRD("-         ",2);
  WITH A DO
  BEGIN
```

Listing 1, continued:

```
    IF RMOO THEN                              (* CASTLE *)
    BEGIN
      ADDWRD("O-O        ",3);
      IF RMQS THEN
        ADDWRD("-O         ",2);
    END
    ELSE                                      (* NOT CASTLE *)
      IF RMCA THEN                            (* CAPTURE *)
        MINGEN(A,SYNCF,SYNCL)
      ELSE                                    (* SIMPLE MOVE *)
        MINGEN(A,SYNMF,SYNML);
    IF RMPR THEN                              (* PROMOTION *)
    BEGIN
      ADDCHR("=");
      ADDCHR(XTGC[RMPP]);
    END;
    ADDWRD(".          ",3);
    IF RMCH THEN                              (* CHECK *)
    BEGIN
      ADDWRD("CHECK      ",5);
      IF RMMT THEN                            (* CHECKMATE *)
        ADDWRD("MATE       ",4);
      ADDCHR(".");
    END
    ELSE
      IF RMMT THEN                            (* STALEMATE *)
        ADDWRD("STALEMATE.",10);
  END;
END;  (* MINENG *)


PROCEDURE MYMOVE;                             (* MAKE MACHINES MOVE *)

VAR
  INRM : RM;                                  (* THE MOVE *)

BEGIN
  CREATE;                                     (* INITIALIZE DATA BASE *)
  INRM := MOVES[SEARCH]:                      (* FIND THE BEST MOVE *)
  IF INRM.RMIL THEN
  BEGIN                                       (* NO MOVE FOUND *)
    GOING := 0;
    IF LSTMV.RMCH THEN                        (* CHECKMATE *)
      WRITELN(" CONGRATULATIONS.")
    ELSE                                      (* STALEMATE *)
      WRITELN(" DRAWN. ")
  END
  ELSE
  BEGIN
    MINENG(INRM," MY MOVE ");                 (* TRANSLATE MOVE TO ENGLISH *)
    WRITELN(MOVMS);                           (* TELL THE PLAYER *)
    THEMOV(INRM);                             (* MAKE THE MOVE *)
    IF SWSU THEN
      WRITELN(BOARD.RBTI,".",NODES," NODES.",BSTVL[AK]);
  END;
END;  (* MYMOVE *)


PROCEDURE YRMOVE;                             (* MAKE PLAYERS MOVE *)

LABEL
  11, 12, 13, 14, 15,                         (* SYNTAX NODES *)
  16,                                         (* SYNTAX ERROR *)
  17,                                         (* AMBIGUOUS MOVE *)
  18;                                         (* NORMAL EXIT *)

VAR
  INTB : TB;                                  (* VALID MOVE FOUND *)
  INTC : TC;                                  (* CURRENT CHARACTER *)
  INTW : TJ;                                  (* MOVES INDEX *)

  INTP : TP;                                  (* MOVING PIECE *)
  INCP : TP;                                  (* CAPTURED PIECE *)
  IFCA : TB;                                  (* CAPTURE *)
  IFPR : TB;                                  (* PROMOTION *)
  IFOO : TB;                                  (* CASTLE *)
  IFQS : TB;                                  (* QUEEN SIDE CASTLE *)
  INTG : TG;                                  (* PROMOTION TYPE *)
  IFMV : TB;                                  (* MOVE FOUND *)

  IFLD : TB;                                  (* R, N, OR B ON LEFT *)
  IFLF : TB;                                  (* K OR Q ON LEFT *)
  IFRD : TB;                                  (* R, N, OR B ON RIGHT *)
  IFRF : TB;                                  (* K OR Q ON RIGHT *)

  INLF : SF;                                  (* FILES ON LEFT *)
  INLR : SR;                                  (* RANKS ON LEFT *)
  INRF : SF;                                  (* FILES ON RIGHT *)
  INRR : SR;                                  (* RANKS ON RIGHT *)

  INRM : RM;                                  (* THE MOVE *)

  FUNCTION NCHIN                              (* DETERMINE IF NEXT INPUT
                                                 CHARACTER IS NOT IN A GIVEN
                                                 SET *)
    (A:SC;                                    (* SET OF CHARACTERS TO CHECK *)
     PROCEDURE YRNXXX)                        (* SEMANTICS ROUTINE TO CALL
                                                 IF NEXT CHARACTER IS IN SET *)
    :TB;                                      (* TRUE IF CHARACTER IS NOT IN
                                                 SET *)

  VAR
    INTB : TB;                                (* SCRATCH *)

  BEGIN
    INTB := NOT (INTC IN A);
    IF NOT INTB THEN
    BEGIN
      YRNXXX;                                 (* EXECUTE SEMANTICS ROUTINE *)
      JNTJ := JNTJ+1;                         (* ADVANCE PAST CHARACTER *)
      WHILE (JNTJ < ZJ)
        AND ((ILINE[JNTJ] = " ") OR (ORD(ILINE[JNTJ]) > ORD(ZC))) DO
        JNTJ := JNTJ+1;                       (* SKIP BLANKS *)
      INTC := ILINE[JNTJ];                    (* NEXT CHARACTER *)
      IF (INTC = ".") OR (INTC = ";") THEN
        GOTO 15;                              (* EXIT SCAN *)
    END;
    NCHIN := INTB;                            (* RETURN TRUE IF CHARACTER IS
                                                 NOT IN STRING *)
  END;  (* NCHIN *)


PROCEDURE YRMHIT;                             (* FOUND A MOVE.  EXITS
                                                 TO AMBIGUOUS MOVE IF THIS
                                                 IS THE SECOND POSSIBLE MOVE.
                                                 SAVES THE MOVE IN INRM
                                                 OTHERWISE.  *)

BEGIN
  IF IFMV THEN GOTO 17;                       (* SECOND POSSIBLE MOVE *)
  IFMV := TRUE;                               (* FIRT POSSIBLE MOVE *)
  INRM := MOVES[INTW];                        (* SAVE MOVE *)
END;  (* YRMHIT *)


PROCEDURE YRMCOM;                             (* COMPARE SQUARES.  CALLS YRMHIT
                                                 IF MOVES[INTW] MOVES THE
                                                 RIGHT TYPE OF PIECE, CAPTURES
                                                 THE RIGHT TYPE OF PIECE, AND
                                                 MOVES TO AND FROM POSSIBLE
                                                 SQUARES *)

BEGIN
  WITH MOVES[INTW] DO
    IF (XTSR[RMFR] IN INLR) AND
       (XTSF[RMFR] IN INLF) AND
       (XTSR[RMTO] IN INRR) AND
       (XTSF[RMTO] IN INRF) AND
       (NOT RMIL) AND
       (BOARD.RBIS[RMFR] = INTP) THEN
      IF RMCA = IFCA THEN
        IF RMCA THEN
          IF RMCP = INCP THEN
            YRMHIT
          ELSE
        ELSE
          YRMHIT;
END;  (* YRMCOM *)


PROCEDURE YRMCAP;                             (* SEMANTICS - CAPTURE *)

BEGIN
  IFCA := TRUE;
END;  (* YRMCAP *)


PROCEDURE YRMCAS;                             (* SEMANTICS - CASTLE *)

BEGIN
  IFOO := TRUE;
END;  (* YRMCAS *)


PROCEDURE YRMCPC;                             (* SEMANTICS - CAPTURED PIECE *)

BEGIN
  CASE INTC OF
   "P": INCP := XTUMP[EP,OTHER[JNTM]];
   "R": INCP := XTUMP[ER,OTHER[JNTM]];
   "N": INCP := XTUMP[EN,OTHER[JNTM]];
   "B": INCP := XTUMP[EB,OTHER[JNTM]];
   "Q": INCP := XTUMP[EQ,OTHER[JNTM]];
  END;
END;  (* YRMCPC *)


PROCEDURE YRMCQS;                             (* SEMANTICS - CASTLE LONG *)

BEGIN
  IFQS := TRUE;
END;  (* YRMCQS *)


PROCEDURE YRMLKQ;                             (* SEMANTICS - K OR Q ON LEFT *)

BEGIN
  CASE INTC OF
   "K": INLF := [F5..F8] * INLF;              (* KING SIDE *)
   "Q": INLF := [F1..F4] * INLF;              (* QUEEN SIDE *)
  END;
  IFLF := TRUE;
END;  (* YRMLKQ *)


PROCEDURE YRMLRB;                             (* SEMANTICS - R, N, OR B ON
                                                 LEFT *)

BEGIN
  CASE INTC OF
   "R":  INLF := [F1,F8] * INLF;              (* ROOK FILE *)
   "N":  INLF := [F2,F7] * INLF;              (* KNIGHT FILE *)
   "B":  INLF := [F3,F6] * INLF;              (* BISHOP FILE *)
  END;
  IFLD := TRUE:
END;  (* YRMLRB *)


PROCEDURE YRMLRK;                             (* SEMANTICS - RANK ON LEFT *)

BEGIN
  IF JNTM = LITE THEN
```

Computer Chess

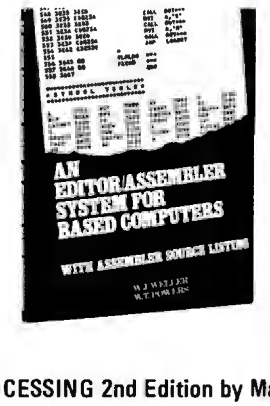
CHESS SKILL IN MAN AND MACHINE edited by Peter W Frey.
☐ A game of endless variations, chess has challenged our skill for centuries. This book surveys our current understanding of human chess skill and covers the subtleties of coaxing a machine to play chess. The initial chapter and appendix present a brief history of the computer chess tournaments. The next two chapters describe the essentials of how humans and computers play chess. The fourth chapter provides a detailed description of the Northwestern Chess Program, currently the national champion. The following three chapters discuss several alternative approaches to chess programming. In the final chapter, a former captain of the U.S. Olympic chess team assesses the present status of chess skill in human and machine. 217 pp. $14.80 hardcover.

CHESS AND COMPUTERS by David Levy.
☐ If you enjoy playing chess, then you will thoroughly enjoy this book, which is loaded with chess games played by computers. The first chapter describes the earliest chess "machine," the famous Automaton chess player that toured Europe and America. There is a detailed account of Torres y Quevedo's invention that played the ending of King and Rook against King. There is also a description of how computers play chess, including an account of early Soviet attempts at chess programming that contains much information hitherto unpublished outside the Soviet Union. Many examples of computer play are given, which provide an excellent insight into the problems facing chess programmers. 145 pp. $8.95.

1975 U.S. COMPUTER CHESS CHAMPIONSHIP by David Levy.
☐ The sixth annual U.S. Computer Chess Championship, held in October 1975, was a tournament in which twelve computer programs competed against each other. This book includes a detailed analysis and description of all the games, presented by David Levy, the tournament director. 86 pp. $5.95.

1976 U.S. COMPUTER CHESS CHAMPIONSHIP by David Levy.
☐ This book includes a detailed analysis and description of all the tournament games played at the seventh annual U.S. Chess Championship held in October 1976. 90 pp. $5.95.

A New Book!



AN EDITOR/ASSEMBLER SYSTEM FOR 8080/8085 BASED COMPUTERS by W J Weller and W T Powers.
☐ This 148-page book contains complete information for initializing and using a powerful new editor/assembler and debugging monitor system, and the full SOURCE text of both. The assembler fully supports all Intel instruction mnemonics as well as the entire language used in PRACTICAL MICROCOMPUTER PROGRAMMING™: THE INTEL 8080. The editor/assembler is resident in less than 8K RAM and will run on any 8080, 8085 or Z80 based computer with peripherals which transfer on a character by character basis or can be made to do so by buffering. The user supplies his own I/O drivers. The text editor is extremely simple to use and does not require irrelevant line numbers. Also included is a program to convert Processor Technology™ format tapes to a format useable by the editor/assembler.

This system is not the usual "quickie" software, riddled with errors and limitations, but a professionally created, thoroughly tested and debugged system. At $14.95 it is the best software bargain you are ever likely to see.

AND BY THE WAY. . . paper tape object copies of all of this software are sent FREE to book purchasers when the coupon at the back of the book is returned to Northern Technology Books. 8½ x 11 format. $14.95.

For your convenience in ordering, please use this page plus the order form on page 138.

DIAL YOUR BANK CARD ORDERS TOLL FREE (800) 258-5477.

BITS™ inc

STANDARD DICTIONARY OF COMPUTERS AND INFORMATION PROCESSING 2nd Edition by Martin H Weik.
☐ This is a very complete, fully cross-referenced dictionary. It goes a step farther in that it includes full explanations, practical examples, many pertinent illustrations, and supplementary information for over 12,500 hardware and software terms. It cross-references the terms to other closely related concepts, and appended to each definition, as the need arises, are explanations, tutorial information, examples, usage areas, and cross-references for further clarification of concepts and meanings. 390 pp. $16.95 hardcover. (No photo)

Circle 35 on inquiry card.

Listing 1, continued:

```
    CASE INTC OF
     "1": INLR := [R1];
     "2": INLR := [R2];
     "3": INLR := [R3];
     "4": INLR := [R4];
     "5": INLR := [R5];
     "6": INLR := [R6];
     "7": INLR := [R7];
     "8": INLR := [R8];
    END
  ELSE
    CASE INTC OF
     "1": INLR := [R8];
     "2": INLR := [R7];
     "3": INLR := [R6];
     "4": INLR := [R5];
     "5": INLR := [R4];
     "6": INLR := [R3];
     "7": INLR := [R2];
     "8": INLR := [R1];
    END;
END;  (* YRMLRK *)


PROCEDURE YRMNUL;                    (* SEMANTICS - NULL *)

BEGIN
END;  (* YRMNUL *)


PROCEDURE YRMPCM;                    (* SEMANTICS - PIECE MOVED *)

BEGIN
  CASE INTC OF
   "P": INTP := XTUMP[EP,JNTM];      (* PAWN *)
   "R": INTP := XTUMP[ER,JNTM];      (* ROOK *)
   "N": INTP := XTUMP[EN,JNTM];      (* KNIGHT *)
   "B": INTP := XTUMP[EB,JNTM];      (* BISHOP *)
   "Q": INTP := XTUMP[EQ,JNTM];      (* QUEEN *)
   "K": INTP := XTUMP[EK,JNTM];      (* KING *)
  END;
END;  (* YRMPCM *)


PROCEDURE YRMPRO;                    (* SEMANTICS - PROMOTION *)

BEGIN
  CASE INTC OF
   "R": INTG := PR;                  (* ROOK *)
   "N": INTG := PN;                  (* KNIGHT *)
   "B": INTG := PB;                  (* BISHOP *)
   "Q": INTG := PQ;                  (* QUEEN *)
  END;
  IFPR := TRUE;
END;  (* YRMPRO *)


PROCEDURE YRMRKQ;                    (* SEMANTICS - K OR Q ON RIGHT *)

BEGIN
  CASE INTC OF
   "K": INRF := [F5..F8] * INRF;     (* KING SIDE *)
   "Q": INRF := [F1..F4] * INRF;     (* QUEEN SIDE *)
  END;
  IFRF := TRUE;
END;  (* YRMLKQ *)


PROCEDURE YRMRRB;                    (* SEMANTICS - R, N, OR B ON
                                        RIGHT *)

BEGIN
  CASE INTC OF
   "R":  INRF := [F1,F8] * INRF;     (* ROOK FILE *)
   "N":  INRF := [F2,F7] * INRF;     (* KNIGHT FILE *)
   "B":  INRF := [F3,F6] * INRF;     (* BISHOP FILE *)
  END;
  IFRD := TRUE;
END;  (* YRMLRB *)


PROCEDURE YRMRRK;                    (* SEMANTICS - RANK ON RIGHT *)

BEGIN
  IF JNTM = LITE THEN
    CASE INTC OF
     "1": INRR := [R1];
     "2": INRR := [R2];
     "3": INRR := [R3];
     "4": INRR := [R4];
     "5": INRR := [R5];
     "6": INRR := [R6];
     "7": INRR := [R7];
     "8": INRR := [R8];
    END
  ELSE
    CASE INTC OF
     "1": INRR := [R8];
     "2": INRR := [R7];
     "3": INRR := [R6];
     "4": INRR := [R5];
     "5": INRR := [R4];
     "6": INRR := [R3];
     "7": INRR := [R2];
     "8": INRR := [R1];
    END;
END;  (* YRMLRK *)
```

```
BEGIN  (* YRMOVE *)
  INTB := FALSE;
  WHILE NOT INTB DO
  BEGIN
    READER;                                   (* READ NEXT MOVE *)
    LSTMOV;                                   (* LIST LEGAL MOVES *)
    IFCA := FALSE;
    IFPR := FALSE;
    IFOO := FALSE;
    IFQS := FALSE;
    IFLD := FALSE;
    IFLF := FALSE;
    IFRD := FALSE;
    IFRF := FALSE;
    INTP := MT;
    INCP := MT;
    INLF := [F1..F8];
    INRF := [F1..F8];
    INLR := [R1..R8];
    INRR := [R1..R8];

    INTC := ILINE[JNTJ];

    IF      NCHIN(["P","R","N","B","Q","K"],YRMPCM) THEN GOTO 14;
    IF      NCHIN(["/"]                    ,YRMNUL) THEN GOTO 11;
    IF      NCHIN(["K","Q"]                ,YRMLKQ) THEN;
    IF      NCHIN(["R","N","B"]            ,YRMLRB) THEN;
    IF      NCHIN(["1".."8"]               ,YRMLRK) THEN;
11:  (* LEFT SIDE DONE *)
    IF NOT NCHIN(["-"]                     ,YRMNUL) THEN GOTO 12;
    IF      NCHIN(["*","X"]                ,YRMCAP) THEN GOTO 16;
    IF      NCHIN(["P","R","N","B","Q"]    ,YRMCPC) THEN GOTO 16;
    IF      NCHIN(["/"]                    ,YRMNUL) THEN GOTO 13;
12:  (* RIGHT SIDE SQUARE *)
    IF      NCHIN(["K","Q"]                ,YRMRKQ) THEN;
    IF      NCHIN(["R","N","B"]            ,YRMRRB) THEN;
    IF      NCHIN(["1".."8"]               ,YRMRRK) THEN;
13:  (* PROMOTION *)
    IF      NCHIN(["="]                    ,YRMNUL) THEN GOTO 15;
    IF      NCHIN(["R","N","B","Q"]        ,YRMPRO) THEN GOTO 16;
    GOTO 15;

14:  (* CASTLING *)
    IF      NCHIN(["O","0"]                ,YRMNUL) THEN GOTO 16;
    IF      NCHIN(["-"]                    ,YRMNUL) THEN GOTO 16;
    IF      NCHIN(["O","0"]                ,YRMCAS) THEN GOTO 16;
    IF      NCHIN(["-"]                    ,YRMCQS) THEN GOTO 15;
    IF      NCHIN(["O","0"]                ,YRMNUL) THEN GOTO 16;
15:  (* SYNTAX CORRECT *)

    IF IFRF AND NOT IFRD THEN
      INRF := INRF * [F4,F5];                 (* SELECT K OR Q FILE *)
    IF IFLF AND NOT IFLD THEN
      INLF := INLF * [F4,F5];                 (* SELECT K OR Q FILE *)
    IFMV := FALSE;                            (* NO MOVE FOUND YET *)
    INTW := AW;                               (* INITIALIZE INDEX *)
    WHILE INTW < JNTW DO
      WITH MOVES[INTW] DO
      BEGIN
        IF RMPR = IFPR THEN
          IF RMPR THEN
            IF RMPP = INTG THEN               (* CORRECT PROMOTION TYPE *)
              YRMCOM                          (* COMPARE SQUARES AND PIECES *)
            ELSE
          ELSE                                (* NOT PROMOTION *)
            IF RMOO = IFOO THEN
              IF RMOO THEN                    (* CASTLING *)
                IF RMQS = IFQS THEN           (* CASTLING SAME WAY *)
                  YRMHIT
                ELSE
              ELSE                            (* NOT CASTLING *)
                YRMCOM;                       (* COMPARE SQUARES AND PIECES *)
        INTW := INTW+1;                       (* ADVANCE MOVES INDEX *)
      END;
    IF IFMV THEN                              (* ONE MOVE FOUND *)
    BEGIN
      MINENG(INRM,"YOUR MOVE ");              (* CONVERT TO OUR STYLE *)
      WRITELN(MOVMS);                         (* PRINT MOVE *)
      THEMOV(INRM);                           (* MAKE THE MOVE *)
      INTB := TRUE;                           (* EXIT YRMOVE *)
    END
    ELSE                                      (* NO MOVES FOUND *)
      WRITELN(" ILLEGAL MOVE.");
    GOTO 18;                                  (* EXIT *)

16:  (* SYNTAX ERROR *)
    WRITELN(" SYNTAX ERROR.");
    GOTO 18;                                  (* EXIT *)

17:  (* AMBIGUOUS MOVE *)
    WRITELN(" AMBIGUOUS MOVE.");
18:  (* EXIT *)
  END;
END;  (* YRMOVE *)



BEGIN  (* THE PROGRAM *)
  WRITELN(" HI.  THIS IS CHESS .5");
  INICON;                                     (* INITIALIZE CONSTANTS *)

1:  (* INITIALIZE FOR A NEW GAME *)
  INITAL (BOARD);                             (* INITIALIZE FOR A NEW GAME *)
  REPEAT
    REPEAT
      YRMOVE;                                 (* EXECUTE PLAYERS MOVE *)
    UNTIL SWRE;

2:  (* EXECUTE MACHINES MOVE *)
    REPEAT
      MYMOVE;
      IF GOING > 0 THEN
        GOING := GOING-1;
    UNTIL GOING = 0;
  UNTIL FALSE;

9:  (* END OF PROGRAM *)
END.
```

Text continued from page 144

A second and somewhat more challenging project would be to develop a transposition table for the program. This requires the availability of unused memory (at least 8 K bytes and preferably 16 K or 32 K bytes), an efficient hashing scheme, and a set of decision rules to select among positions when a collision occurs (ie: two positions hash to the same address in the table). Another problem is that the use of a staged evaluation process and the α-β algorithm often provides an imprecise evaluation score (ie: the machine has determined that a position was not optimal but has not invested the time to find out exactly how bad it was). If the programmer succeeds with the transposition table, however, move calculation will take 30 to 50 per cent less time in most middle game positions and 60 to 90 per cent less time in many end game positions.

A third area for improvement is the evaluation function. Our program presently has only a rudimentary function. The reader should compare it with the one used by Chess 4.5 which is described in detail by Slate and Atkin. Their evaluation function provides an excellent starting point for revising our present function. In part 4 we will discuss the advantages of using a conditional evaluation function, ie: one that changes depending on the stage of the game and on the presence of special features. One implementation of this strategy is the special end game program described by Monroe Newborn in *Chess Skill in Man and Machine.*

It is appropriate for us to add two important disclaimers at this juncture. Although we have carefully tested each of the routines in the program and played several chess games, it is still possible that there are a few minor bugs in the program. If you find one, a letter to one of us or to BYTE would be appreciated. Secondly, our chess program was written primarily for pedagogical purposes. For this reason it is not a production program and does not run very efficiently. If you are the competitive type, our program should provide many useful ideas, but you should not expect it to compete successfully in tournament play unless you make extensive modifications and additions.

A chess program has a tendency to grow and change its personality as the programmer becomes more familiar with each of its many limitations. It provides a constant challenge for those of us who are too compulsive to tolerate obvious weaknesses. In fact one must be careful not to become totally obsessed with this project. We do not wish any of you to lose your job or your spouse because of a chess program.■

Continued from page 10

recovery from transmission errors. Add to this the fact that the protocol has been in service a number of years, and I am sure readers will find the literature worth reviewing.

Carroll Perkins
POB 333
Pilot Mountain NC 27041

SIZING UP MODULAR PROGRAMMING

I enjoyed the "Top-Down Modular Programming" by Albert D Hearn in the July 1978 BYTE, page 32. I thought he did a good job of explaining the subject. While I realize that he was purposely trying to simplify matters, I do take exception to his comment that a module should be no more than 50 lines long.

The concepts of structured programming are intended as guidelines, not as the dogma for a programmer's religion. All of the better known proponents of the methodology stress this point, along with the idea that you must approach the study of structured programming with your eyes open, making your own evaluation. In this light let us explore the 50 line limit.

One of the bases for breaking a program up into modules is so that a complex problem can be handled with *small*, easy to understand pieces of code. One of the thoughts about module size is, therefore, that a module ought to be able to fit on one printed page. This is so that all the information about the module is in one place and the programmer won't have to thumb through several pages to read the code for a single module. Having experienced "modules" running as long as 10 to 15 pages, I heartily agree with this philosophy.

In professional programming installations, this idea has frequently been translated into a local standard of about 50 lines of code, since this is the number of lines which are printed on an 8.5 by 11 inch (21.59 by 27.94 cm) page coming out of a line printer (allowing for headers, footers, etc). For the personal computer enthusiast, however, this limit might be more conveniently set at 24, 32 or 40 lines—the size of the video display.

For many more complex problems, it is possible that a significant module cannot be constructed in 24 lines. This is no problem—just make the modules longer. The point is to try to restrict the module size to a length which enhances the programmer's ability to understand the code.

James Fleming
2220 Sims Dr
Columbus IN 47201

LIGHT SEEKING ROBOT

I enjoyed your "On Building a Light-Seeking Robot Mechanism" article (August 1978 BYTE, page 24). The sonar description caught my eye, since I am working on a similar problem, and the approach in the article appears to be more complex than necessary. $T_O + T_L$ and $T_O + T_R$ describe a unique point in the plane. Only if human interaction with this data were intended would it be necessary to translate this to θ and T_O. An array in memory could map the plane, or decision points could be established and checked against current values. We must be careful to avoid limiting the robot to human perceptions.

John Gledhill
678 Washington Av #4
Yuba City CA 95991

CIARCIA'S CIRCUIT CELLAR

I am not one for writing to a magazine, but your article "Let Your Fingers Do the Talking" (August 1978 BYTE, page 156, and September 1978 BYTE, page 94) was right on! I read magazines and use the ideas and never let editors know. I sometimes write the authors.

Mr Ciarcia's article was just the best idea for a teacher like me.

I have been trying to get such an idea and equipment for my kids in my classroom.

This article has helped me take one more step toward my ideal computer learning system for my kids. The touch panel is a great help for younger kids. Thanks!

I will be busy building the hardware and developing the software. I can't wait for the next article. Tell Steve to keep the software coming and tell him what a great contribution he has made. I can't wait to see my kids' reaction.

Harold R Whitlock
616 2nd St
White Bear Lake MN 55110

A QUESTION OF PATENTS

I read with interest the articles entitled "Let Your Fingers Do the Talking," as they relate to a noncontact touch scanner or panel. As you are aware, the University of Illinois has two patents covering the touch panel described in the article. They are US patent numbers 3,775,560 and 3,860,764. My company represents the University of Illinois Foundation in the licensing of these patents.

While we have no objection to your description of how to build the touch panel, I would appreciate your taking appropriate steps to notify BYTE readers that, if they desire to build the touch panels commercially, they will need a license under our patents.

These patents have already been licensed to several companies in the US and their foreign counterparts licensed abroad. We are willing to provide licensing terms and conditions to interested commercial entities.

Thank you for your consideration.

A Sidney Alpert
University Patents Inc
2777 Summer St
Stamford CT 06905

Steve Ciarcia replies:

While I am personally aware of journalistic freedoms with regard to patents, many readers may not be. Your statement is well taken. It should be further noted that the University of Illinois patents appear to cover the scanning principle and not the design circuitry.

PLATO AND THE TOUCH PANEL

I lead a double life: during the night I make up little things for my own Apple, but during the day I become a rising young training executive of the CDC Plato terminal.

And that's why Steve Ciarcia's article on touch input units brought me up with a start — because his touch panel is nearly identical to the touch panels put on the first Plato terminals! I'm taking about the original Magnavox terminals that brought Plato out of the lab in Illinois and into the world. Those terminals, just like Steve's monitor, had a picture frame around them with 32 LEDs and 32 phototransistors. In fact, there are only minor differences between them and Steve's design except for the aspect ratio — Plato terminals have square screens. The Plato panels even had a built-in circuit that triggered the 0.1 second beep Steve mentioned in his article.

One minor difference was that Plato panels had the LEDs on the top and the phototransistors on the bottom. Steve's method is better — the old touch panels would fail regularly whenever the sun shone on them at the wrong angle.

Those old panels are obsolete now. Both CDC and the other Plato terminal maker are using different designs, under direct control from an internal microprocessor. But there are still plenty of the old Magnavox boxes out there, in schools and colleges across the country. In fact, three microprocessor systems that I know of display output through a gutted and rewired Magnavox box, bought on the "orange and black market." (named for the color of the old plasma screens, you know.)

Silas S Warner
8 Charles Plaza
Baltimore MD 21201

Steve Ciarcia replies:

Thank you for the vote of confidence. It may interest you to know that I worked for Control Data Corporation for three years as a process control engineering consultant. During that time I became familiar with Plato, but never have, to this day, used the touch input feature. I just liked to go in after work and play Star Trek or Empire against other people on the system throughout the country.

The design illustrated in BYTE was done from scratch and any resemblance to Plato is purely coincidental. I just got an idea for it one Friday afternoon and brought the completed unit into the office Monday morning.

VOICING AN OPINION

Congratulations to Bill Georgiou for his excellent primer on speech recognition in June 1978 BYTE ("Give an Ear to Your Computer," page 56). This wide-ranging and complex topic was presented in a most understandable form, yet did not sacrifice excessive detail.

As Mr Georgiou stated, voice recognition has a rather long history, and has intrigued avid experimenters such as myself for some time. Back in 1965, I designed and built a demonstration unit capable of differentiating ten different words or short phrases, and activating one of ten relays upon completion of analysis. The project was awarded a first prize at the Canada-Wide Science Fair that year.

The implementation was not unlike figure 12 in the article, an automatic gain control stage followed by multiple bandpass filters, except that all processing was done in analog. A degree of differential comparison was incorporated, to provide for the variableness of fundamental pitch in each speaker. Template matching was used, with a great deal of "cut-and-try" programming effort. And if Mr Georgiou thinks the Bell Labs version of 1952 was "grotesque," he should have seen this one, built from old television sets and record players.

Although the machine displayed about a 90 percent recognition rate for my voice (it had obviously been programmed that way), I was constantly surprised during public demonstrations how often it would react correctly to a "stranger's" voice. With a little practice, even a feminine voice could speak the word "open" and see my little solenoid lock snap back.

The article has rekindled my interest in the field, and I shall be looking forward to implementation with my microprocessor now.

F Wallace
Burroughs Business Machines Ltd
POB 861
Winnipeg, Manitoba
CANADA R3C 2P7

CALLING ALL COMPUTERS

I got quite a zing out of Donald Newcomb's letter on the evils of radio communications between computers, having just read the DOC's (the Canadian FCC) proposals on the "Packet Radio Service." Welcome to the 19th century, Mr Newcomb!

The DOC is proposing not 1 MHz, but the entire amateur radio 220 MHz band and very likely will finalize at 220-221 MHz, leaving room for a communicator class or GRS (CB) service in the remaining 4 MHz. Our FCC, which works very closely with the DOC, is thinking along the same lines.

Unquestionably, we'll have radio communications between computers within the next year or two. I, for one, am extremely excited by the fantastic possibilities, and am already working on suitable equipment. I'd appreciate hearing from others working along the same lines.

Donad L Stoner W6TNS/7
John Hancock Bldg
Mercer Island WA 98040

IDEAS NEEDED FOR PROJECT TO AID DISABLED

The Spain Rehabilitation Center at the University of Alabama Medical Center has a project underway to demonstrate both the utility and economic feasibility of the new generation of personal computers for use by the severely disabled. The programmability of the computer will allow it to serve as a general purpose appliance to be used as an aid to communication and education as well as for environmental control and entertainment.

This system, as currently envisioned, will consist of a microcomputer; an on line storage device for programs and data; two television monitors for user feedback and information display; a printing device for typed output; a speech recognition device for vocal input of commands, data, and text; a power line controller for environmental control; and a telephone dialing and answering device. We are attempting to select components which are widely distributed and serviced as well as being plug compatible and economically priced.

Programs will be written or purchased to perform specific functions in each of the four general areas mentioned above. However, we would be very interested in receiving ideas from your readers, particularly those who are disabled, those who have disabled friends or relatives, and those who have personal computers and would like to develop hardware or software for the system on their own, regarding specific functions which they would like to see developed and which could be accommodated by the proposed microcomputer system.

We are looking forward to receiving input from anyone who may be interested in this project.

Charles Healey
Spain Rehabilitation Center
UAB University Sta
Birmingham AL 35294■

BYTE's Bits

Pascal for Computer Club Members

The UCSD Pascal Project has announced a special offer for bona fide computer clubs. The UCSD Pascal software, which normally sells for $200, will be made available to club members at a sustantial discount if the club assists in the copying and production costs for disseminating the software. For more information, computer clubs should contact Tracy Barrett, CO 21, UCSD, La Jolla CA 92093.■

Attention: HAL 9000 Owners

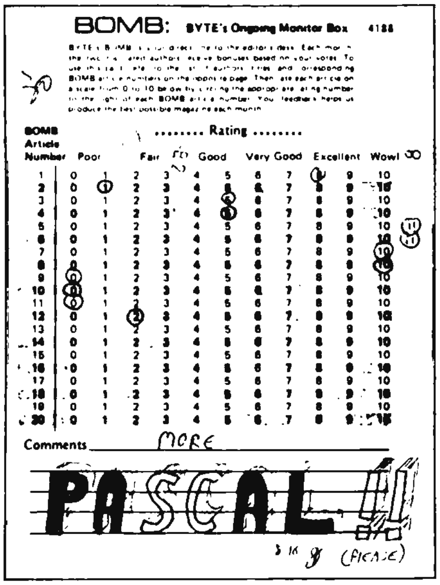
We have just received word of an important new book: *Programming Instructions for the HAL 9000 Computer,* revised edition. The new edition of the HAL 9000 handbook has been updated to incorporate improvements suggested by this versatile machine's surviving users. In particular, the manufacturers suggest that priority be given to the retrofitting of small explosive charges at strategic points in the central memory unit.

Authorized by
Arthur C Clarke
Chief Semantic Controls Engineer
Colombo SRI LANKA■

Some BOMB Reflections

Occasionally we like to share some of the unfettered comments, pro and con, that arrive monthly on the BOMB cards, our system for reader response through a postcard at the back of the magazine.

The following BOMB card came from an anonymous Pascal enthusiast:



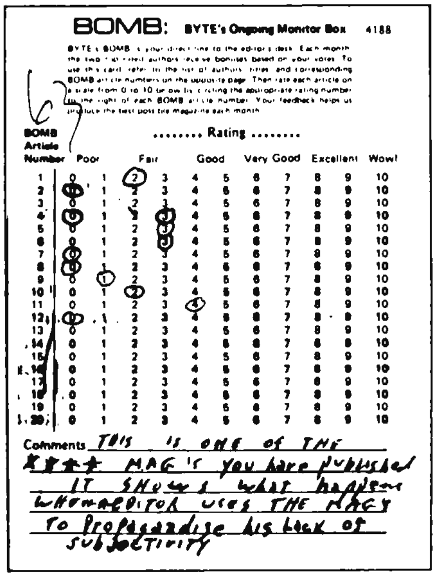
Other BOMB comments about the August issue include:

Best BYTE I have read in a long time. Please devote more discussion to Pascal. The language in its beauty, compactness and readability is worth talking about. Would like to know more about the extensions being discussed for the language.

The article by Weems ("Designing Structured Programs") was easy to comprehend and delivered a valuable message.

This was one of your best issues. I like having related articles in one issue.

To be fair, not all the comments were as positive as the last three. The following BOMB card was also received for the month of August:



We have been suitably chastened. This is the first time we have been accused of not being subjective enough, and we will attempt to examine the problem as subjectively as possible. . . .CM■

Incremental Motion Control Symposium Issues a Call for Papers

The Eighth Annual Symposium on Incremental Motion Control Systems and Devices will be held at the Ramada Inn, Urbana IL, May 21 thru 24 1979. A call for papers has been issued by Prof B C Kuo, director of the symposium, which is sponsored by the Incremental Motion Control Systems Society, in cooperation with the University of Illinois, Dept of Electrical Engineering, and Warner Electric Brake and Clutch Company of Beloit WI.

The symposium will encompass a broad area with sessions devoted to tutorial papers as well as original contributions, covering step motors,

You Are What You Eat

machine tool control systems, computer controls, linear and AC/DC motors, clutch-brake devices and systems, and related incremental motion control applications. Exhibit space will be available.

The call for papers, both theoretical and practical, related to incremental motion control systems and devices, requests authors to submit a summary of about 500 words by January 1. Final manuscripts will be due around mid March. Send all summaries and inquiries to Dr B C Kuo, POB 2772, Station A, Champaign IL 61820, phone (217) 333-4341.■

A Call for Papers

The Instrument Society of America has issued a call for papers for its 1979 conference and exhibit, ISA/79. The conference will take place at the O'Hare Exposition Center in Chicago IL, from October 22 through 25 1979. The conference theme, Instrumentation for Energy Alternatives, will emphasize current practices in instrumentation design and implementation. Papers are being sought in the following areas: analysis, automatic control, chemicals and petroleum, cryogenics, data handling and computation, education and training, food, glass and ceramics, maintenance, management, marine sciences, metals, mining and metallurgy, power, process measurement and control, standards and practices, telemetry, test measurement, water and wastewater, pulp and paper, and biomedical instrumentation. Paper topics should introduce or explain techniques or innovations in instrumentation and control systems design, testing, operation and maintenance. The papers may be either theoretical in nature or application oriented. The deadline for unsolicited abstracts is February 5 1979. The appropriate forms should be requested from: ISA/79, Instrument Society of America, 400 Stanwix St, Pittsburgh PA 15222.■

American Management Association Offers Courses for EDP and NonEDP Professionals

The American Management Association's Information Systems and Technology Division is offering courses for both the electronic data processing (EDP) and nonEDP professional managers. A sampling of the courses offered include: systems analysis and design computer security, minicomputers, distributed data processing, fundamentals of EDP, office automation, EDP applications to human resources and much more. The courses run through March 1979. A brochure describing these courses is available from the American Management Association, 135 W 50th St, New York NY 10020.■

Addendum

In the September 1978 BYTE we published a write-up in the "What's New?" section on an assembly language development system for 8080 and Z-80 microcomputers (page 198). We would like to pass along some additional information, which we have just received: the PDS development system is available for $99 and is operational on any Z-80, 8080, and 8085 computer using the North Star or Micropolis Model II disk units. Contact Allen Ashley, 395 Sierra Madre Villa, Pasadena CA 91107.■

BYTE's Bugs

Chess Bug

Due to an oversight on our part, a number of errors appeared in the October 1978 "In This BYTE" entry (page 4) for "Creating a Chessplayer" by Peter W Frey and Larry Atkin. The introduction implies that Peter Frey is one of the creators of Chess 4.6. This is not true. The program was written by Larry Atkin and David Slate. Peter Frey was also erroneously referred to as "David" Frey, and Larry Atkin as Larry "Atkins." We sincerely apologize to all concerned for these errors.■

Address Change

In the "What's New?" section of November 1978 BYTE we reported on a new product from TSA Software (page 216). This company has notified us of a change of address. Their new address is: 39 Williams Dr, Monroe CT 06468.■

Tilt!

Some errors crept into the Pinball Wizard program that appeared in my article "The HP-67 and HP-97: Hewlett-Packard's Personal Computers" (June 1978 BYTE, page 112). The code in step 69 of listing 1 (page 114) should read "35 00," not "36 00"; the key entry is correct, however. Also, step 189 (page 115) should read "RCL B 36 12," not "RCL 6 36 06," and step 139 should read LBL 0 21 00 instead of LBL 0 21 16.

My apologies to any readers who might have experienced difficulties in using the program.

Craig A Pierce
2529 S Home Av
Brewyn IL 60402■

Event Queue

In order to gain optimum coverage of your organization's computer conferences, seminars, workshops, courses, etc, notice should reach our office at least three months in advance of the date of the event. Entries should be sent to: Event Queue, BYTE Publications, 70 Main St, Peterborough NH 03458.

December 3-5, Ninth North American Computer Chess Championship, Sheraton Park Hotel, Washington DC. The 1978 annual meeting of the Association for Computing Machinery will be the site of this chess championship. This will be a 4 round Swiss style tournament with participants restricted to computers. Two rounds will be played on December 3 (1 PM and 7:30 PM), one on Monday (7:30 PM) and the last round on Tuesday (7:30 PM). Deadline for entries is October 20. Contact Prof M M Newborn, School of Computer Science, McGill University, Montreal Quebec H3A 2K6, CANADA.

December 4-6, 1978 Annual Conference of the Association for Computing Machinery, Sheraton Park Hotel, Washington DC. Contact Dr Richard Austing, Dept of Computer Science, University of Maryland, College Park MD 20742, (301) 454-2004.

December 4-6, Minicomputers and Distributed Processing, Atlanta GA. This 3 day seminar will examine the uses, economics, programming and implementation of minicomputers. Contact Philip M Kowlen, director, Center for Continuing Education, The University of Chicago, 1307 E 60th St, Chicago IL 60637.

December 4-8, Microcomputer Software Design, Virginia Polytechnic Institute and State University, Blacksburg VA. This workshop will develop assembly language programming skill for 8080 and 8085 based microcomputers. Topics to be discussed include floating point mathematics, lookup tables, number base conversion, interrupt programming, searching and sorting. Contact Dr Linda Leffel, Donaldson Brown Center for Continuing Education, Virginia Polytechnic Institute and State University, Blacksburg VA 24061, (703) 951-5421.

December 6-8, Data Processing Operations Management, Washington DC. This 3 day seminar will emphasize the management skill and techniques applicable to the data processing operations function. Contact Phillip M Nowlen, director, Center for Continuing Education, The University of Chicago, 1307 E 60th St, Chicago IL 60637.

December 12-14, Midcon/78, Dallas Convention Center, High Technology electronics show and convention. Contact Electronic Conventions Inc, El Segundo CA, (800) 421-6816 (toll free).

December 13, Computer Networking Symposium. Sponsored by the IEEE Computer Society's Technical Committee on Computer Communications and the Institute for Computer Sciences and Technology of the National Bureau of Standards. This symposium will highlight papers of practical and research experiences concerning both computer and communication networks. Contact Dr George Cowan, Computer Sciences Corp, 6565 Arlington Blvd, Falls Church VA 22046.

December 13-15, Distributed Minicomputer Networks, Executive Tower Inn, Denver CO. This seminar will address the minicomputer from the viewpoint of the distributed network user. The structure and management of a large data base and software problems with the trade-offs of languages utilized, hardware types, input and output options, device controllers, system failure and recovery, sample application case studies and the economics of minicomputer applications will be covered in depth. Contact The Institute for Professional Education, Suite 601, 1901 N Fort Myer Dr, Arlington VA 22209, (703) 527-8700.

December 18-21, Microcomputer Data Acquisition, Instrumentation and Measurement Systems, Virginia Polytechnic Institute and State University. Course to be given by the authors of the *Bugbooks.* Contact Dr Linda Leffel, Donaldson Brown Center for Continuing Education, Virginia Polytechnic Institute and State University, Blacksburg VA 24061, (703) 951-5421.

January 8-12, Structured Programming and Software Engineering, George Washington University, Washington DC. This course is designed for experienced program architects, designers and managers. It will provide up to date technical knowledge of logical expression, analysis and invention for performing and managing software architecture, design and production. Presentations will cover principles and applications in structured programming and software engineering. Design workshops with analysis and review sessions will provide actual practice in problem solving. Contact George Washington University, Continuing Engineering Education, Washington DC 20052.

January 15-17, Minicomputers and Distributed Processing, San Francisco. For details, see December 4-6, Atlanta.

January 17-19, Distributed Minicomputer Networks, Ramada Inn, Arlington VA. For details, see December 13-15, Denver.

January 24-27, International Microcomputers/Minicomputers Microprocessors '79/Japan, Harumi Exhibition Center, Tokyo. Contact ISCM, 222 W Adams St, Chicago IL 60606, (312) 263-4866.

January 30-February 1, Communication Networks Conference and Exposition, Sheraton Park Hotel, Washington DC. Designed to bring together communication network users, consultants, vendors and regulatory officials so that issues can be discussed and analyzed. It is particularly aimed at executives and managers who purchase communication products and services. Contact The Conference Company, 60 Austin St, Newton MA 02160.

February 1-3, Microprocessor Programming Workshop with a Take-Home Microprocessor, Jefferson Plaza Building, Arlington VA. Sponsored by the IEEE, this 3 day workshop is intended for the practicing engineer, engineering manager and programmer. The course objective is to provide state of the art information in order to acquire an understanding of the place of microprocessors as replacements for wired logic and as controllers; to provide the capability of understanding the design of systems involving microprocessors; and the ability to program the Motorola M6800 microprocessor in machine language. All students will have their own microprocessor and laboratory equipment. Contact IEEE Service Center, 2145 Hoes Ln, Piscataway NJ 08854.

February 13-15, The National Office Exhibition and Conference, Harbour Castle Hilton Convention Center, Toronto Ontario. This 3 day exhibition will provide a showplace for approximately 100 exhibitors in the areas of word processing, office computers, office equipment and furniture. Contact *Canadian Office* magazine, 2 Bloor St W, Suite 2504, Toronto Ontario, CANADA M4W 3E2, (416) 967-6200.

February 14-16, The IEEE International Solid-State Circuits Conference, Philadelphia PA. Forum for the presentation of new advancements in all aspects of solid-state circuits. Contact Lewis Winner, 301 Almeria Av, POB 343788, Coral Gables FL 33134.■

Clubs and Newsletters

Attention: Robot Enthusiasts

The Robotics Newsletter is a monthly periodical for robot enthusiasts on both the hobby and professional level. It presents timely articles on microcomputers, batteries, motors, automata theory, sensory devices, manipulators, biophysical analogs, robot history, etc. Yearly subscriptions are $8. Contact the International Institute for Robotics, POB 615, Pelahatchie MS 39145.

Northern New England Computer Society

Albert Brunelli has written us that a new computer club has been formed "up here in the north woods." It is located in Berlin NH and is called the Northern New England Computer Society. They meet the second Thursday of each month at 7 PM at the New Hampshire Vocational Technical College, Milan Rd, Berlin. Their aim is to set up an area that is accessible to local people where they can learn about and use small computers. The membership fee is $10 per year. For more information, write to Albert Brunelli at POB 69, Berlin NH 03570.

Denver Amateur Computer Society

The Denver Amateur Computer Society now has permanent club quarters and office at 1380 S Santa Fe Dr, Denver CO 80223. The club meets the third Wednesday of every month at 7:30 PM. For further information, write to the society's president, Mike Dymtrasz, at the above address or call (303) 979-6441.

The Okaloosa Computer Hobbyist Club

We have been notified that the Okaloosa Computer Hobbyist Club has been formed in Ft Walton Beach FL. The meetings are held on the second and fourth Tuesday of each month and all interested persons and newcomers are welcomed. For more details, write to Loretta R Guske, #72, 32 Denton Blvd NW, Fort Walton Beach FL, (904) 242-5938.

Computers in Mental Health Newsletter

Micro-Psych is a newsletter for professionals interested in the use of computers in mental health. Each bimonthly edition contains reviews of current work in the field, a forum for the exchange of information, an ongoing bibliography, and news about pertinent hardware and software. Membership and a subscription to *Micro-Psych* costs $10 per year. For more information about this newsletter, contact Marc D Schwartz, MD, 26 Trumbull St, New Haven CT 06511.

Connecticut Computer Club Welcomes All Level Hobbyists

The Connecticut Computer Club, which is a few years old, consists of an informal group of software and hardware people who meet on a monthly basis to exchange ideas. Speakers and demonstrations are of general group interest. The club meets the first Thursday of each month at either the Suffield Library or The Computer Store of Windsor Locks. A newsletter is available to members at a yearly cost of $5. Contact Leo Taylor, 18 Ridge Ct W, W Haven CT 06516, (203) 933-5918.

Quad City Computer Club

We have heard from John E Greve, president of the Quad City Computer Club (QC^3). The club, devoted to all phases of hobbyist computing, meets on the first Sunday of each month at 7 PM at the Rock Island Arsenal classroom #5, Rock Island IL. The dues are $6 per year, which includes a newsletter. For more information concerning this club, contact John E Greve, 4211 7th Av, Rock Island IL 61201.

Apple II Users Group in Portland OR

The Apple Portland Program Library Exchange, or APPLE for short, has been formed as a users group for owners of the Apple II. They are interested in exchanging programs and ideas with other clubs. Send a self-addressed stamped envelope for an application form to Ken Hoggatt, 9195 SW Elrose Ct, Tigard OR 97223.

Canadian PET Owners Start Users Group

The Vancouver PET Users Group recently held their second meeting. Attendees included 40 owners and 15 PETs. The club format includes a short presentation by a PET owner on programming, or PET hardware news from Commodore and other sources. This is followed by PET patter and program swapping. For more information about this club, write to Richard Leon, Vancouver PET Users Group, POB 35353, Station E, Vancouver British Columbia, CANADA.

TRS-80 Computing

A complimentary copy of *TRS-80 Computing* has been sent our way. This 32 page first edition is packed with TRS-80 news including articles by a TRS-80 designer, a Radio Shack repairman, and a couple of programmers; an article on how to install your own 16 K byte integrated circuits and save; a 10 page schematic, and considerably more. The price for 12 issues is $10 and is available from Computer Information Exchange Inc, POB 158, San Luis Rey CA 92068.

Chicago Area Computer Hobbyist Exchange Forms User Groups

In keeping with the club's philosophy of dedication to investigating the roles and uses of microcomputers and related small size computing devices in the hobbyist field, the Chicago Area Computer Hobbyist Exchange (CACHE) has recently announced the formation of user groups. According to the club's newsletter *CACHE Register*, the North Star, SOL and Digital Group user groups are active and going strong. The CP/M, H-11/LSI-11 and computer aided instruction user groups have formed but are not meeting regularly. There are other groups in the formative stages. CACHE members meet on the third Sunday of each month at 1 PM in the Northern Illinois Gas Building, Golf and Shermer Rds, Glenview IL. Contact Bill Precht, president, POB 52, S Holland IL 60473.

Attention Minneapolis/St Paul Apple II Users

We have been notified by Dan Buchler, 13516 Grand Av S, Burnsville MN 55337, that an Apple II users group has been formed in the Minneapolis and St Paul area. The purpose of the group is to promote the exchange of user developed programs and technical information among Apple II users. Help in documenting programs will be offered. Contact Dan Buchler for further information.

Newsletter for Users of Digital Group Equipment

BRIDGE (Bidirectional Reflections for the Illumination of Digital Group Enthusiasts) is an impressive newsletter devoted to helping fellow Digital Group owners over the voids. The cost of membership is $6, which entitles you to six issues of the newsletter. The most recent newsletter contains a couple of articles, items for sale, random bits of information of interest to Digital Group equipment owners and much more, including a letter from BYTE's Steve Ciarcia, an occasional contributer. If this newsletter is of interest to you, it can be obtained by writing to the Digital Group Independent Users Group, POB 316, Woodmere NY 11598.

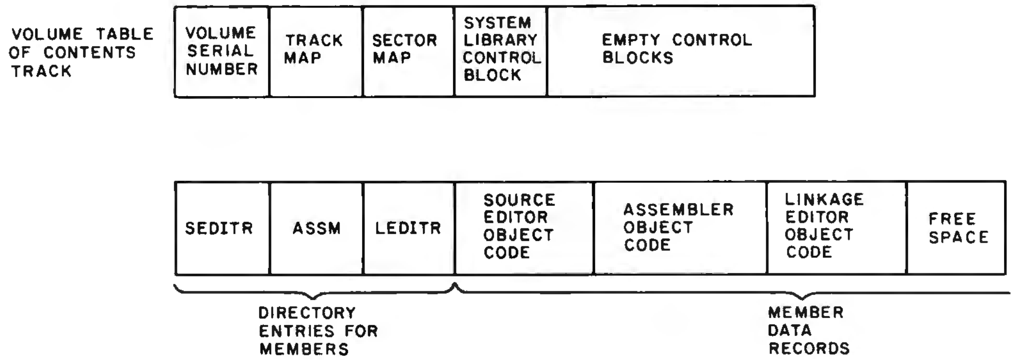
Rockwell's AIM 65 Users Group

An AIM 65 users group is being formed for Rockwell's computer-on-a-board. A bimonthly newsletter will be available in January 1979 for $5 per year. Article contributions are welcome. For more information about this group, contact Don Clem, RR #2, Spencerville OH 45887. ■

Partitioned Data Sets

A I Halsema
32014 Grenville Ct
Westlake CA 91361

Figure 1: Information arrangement for a small partitioned data set.



If you have a floppy disk and are having trouble keeping track of where your programs and data are written on it, this simple file organization technique may provide the automated management of disk space you need.

A partitioned data set (PDS) is a file divided (or partitioned) into areas, each area containing data not related to data in other areas. For example, a system library might contain a source editor, assembler and linkage editor. Each of these programs could be stored in a separate partition in a partitioned data set. The partitioned areas in which these programs would be stored are called members of the data set, so the partitioned data set just described would contain three members, as in figure 1.

About the Author

Mr Halsema is project leader of a funds transfer system for a subsidiary of the First National City Bank. At home he owns a SwTPC 6800 with cassette storage and video terminal.

(1a)

Volume label	6 bytes
Number of table of contents sectors	1 byte
Tracks per volume	1 byte
Sectors per track	1 byte
Bytes per sector	2 bytes
Track map	10 bytes
Sector map	97 bytes
Unused	10 bytes

(1b)

Data set name	6 bytes
Beginning address of data set	2 bytes
Ending address of data set	2 bytes
Length of logical records in data set	1 byte
Length of directory in sectors	1 byte
Unused	4 bytes

Table 1: Possible arrangement for the label record (1a) and data set control block (1b) for a partitioned data set.

Designing the Partitioned Data Set

Four things are required for definition of a partitioned data set. First, a map for defining those areas on the disk that are in use (allocated). For this we create a track map that defines each track on the disk with one bit. If the bit is set to 1, the corresponding track is free. If the bit is set to 0, the track is in use. For a disk with 77 tracks, a 10 byte track map is sufficient. The position of each bit in the map defines the address of its associated track. The first bit in the map defines track 0, the second bit defines track 1, etc. The track map is referenced whenever a data set is allocated or scratched, so the smallest data set possible would be 1 track in length.

Second, we need a sector map to keep track of which sectors are in use and which are free. As in the track map, we assign 1 bit in the map to each sector. If our disk has 10 sectors per track, the sector map must be 770 bits in length, so we assign 97 bytes to it. As in the track map, the position of each bit defines the address of the associated sector. The sector map, table 1, is used when a member is created in or deleted from a data set.

Third, a control block for defining the name and location of the partitioned data sets is needed. These blocks can immediately follow the maps, and should contain space for the data set name, starting track address of the data set, length of directory in tracks, and the ending track address of the data set. Following this control block are similar

Member name	6 bytes
Start of data area address for member	2 bytes
End of data area address for member	2 bytes
Number of logical records in use	2 bytes
Data type of member	1 byte
Unused	3 bytes

Table 2: Format for the directory entry information.

control blocks for other data sets and unused (free) control blocks. A free control block is indicated by the name field being filled with binary 0s.

The maps and the control blocks can all fit on a single track. This track is called the volume table of contents (VTOC) and begins with the volume label, also known as the volume serial number (VSN). If we want a multitrack volume table of contents, we need to define how many tracks are in the table for use by the access method software. A byte for a count of the tracks in the volume itself should be included. If our system is to handle different formats and densities it would be wise to include the format information in the table of contents. The access method software could then read the count of the number of tracks on the disk and the number of sectors per track directly from the table and be able to handle several formats without modification.

In order to avoid wasting disk space, 16 byte logical records can be blocked 8 to a 128 byte sector. A single track volume table of contents blocked in this way could handle 136 partitioned data sets. The fourth item required for a partitioned data set is a directory for the members in the data set. The directory, table 2, begins at the first sector of the first track of the actual data storage area. The directory entries are 16 bytes long and packed 8 to a 128 byte sector. Each entry contains the name of the member, the starting and ending track and sector addresses of the member, the count of the number of sectors actually used by the member, and the data type of the member. The data type may be:

0 : Source data.
1 : Core image object data.
2 : Relocatable object data.

Both source and BASIC programs may be stored in a member of data type 0, while data type 2 is used as input to a relocating loader, linking loader or linkage editor. Data type 1 is used for storing nonrelocatable programs.

Now that the design of the file structure is complete, we can design software that will create, manipulate and delete members and data sets.

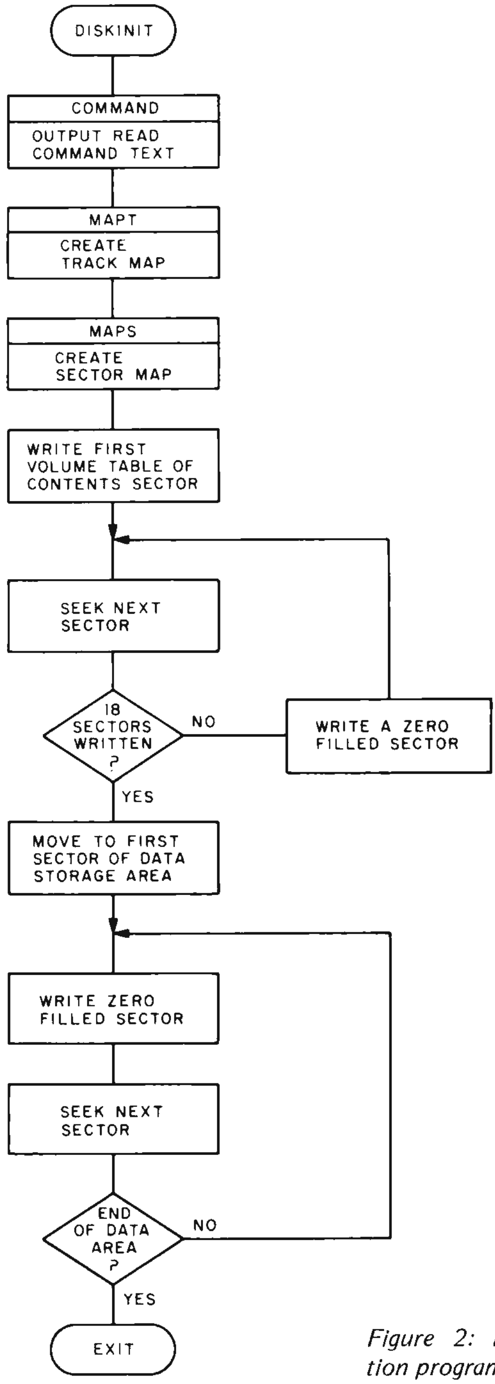


Figure 2: Disk initialization program.

Implementation

The first program we need is a disk initialization routine such as the one in figure 2. The initialization routine creates and empties the volume table of contents track and clears the data storage area by setting it to 0. The user is required to enter the volume serial number and the beginning and ending track numbers of the data storage area. The track and sector maps are created so that the bits corresponding to the data storage area are set to 1 (free space) and all other bits

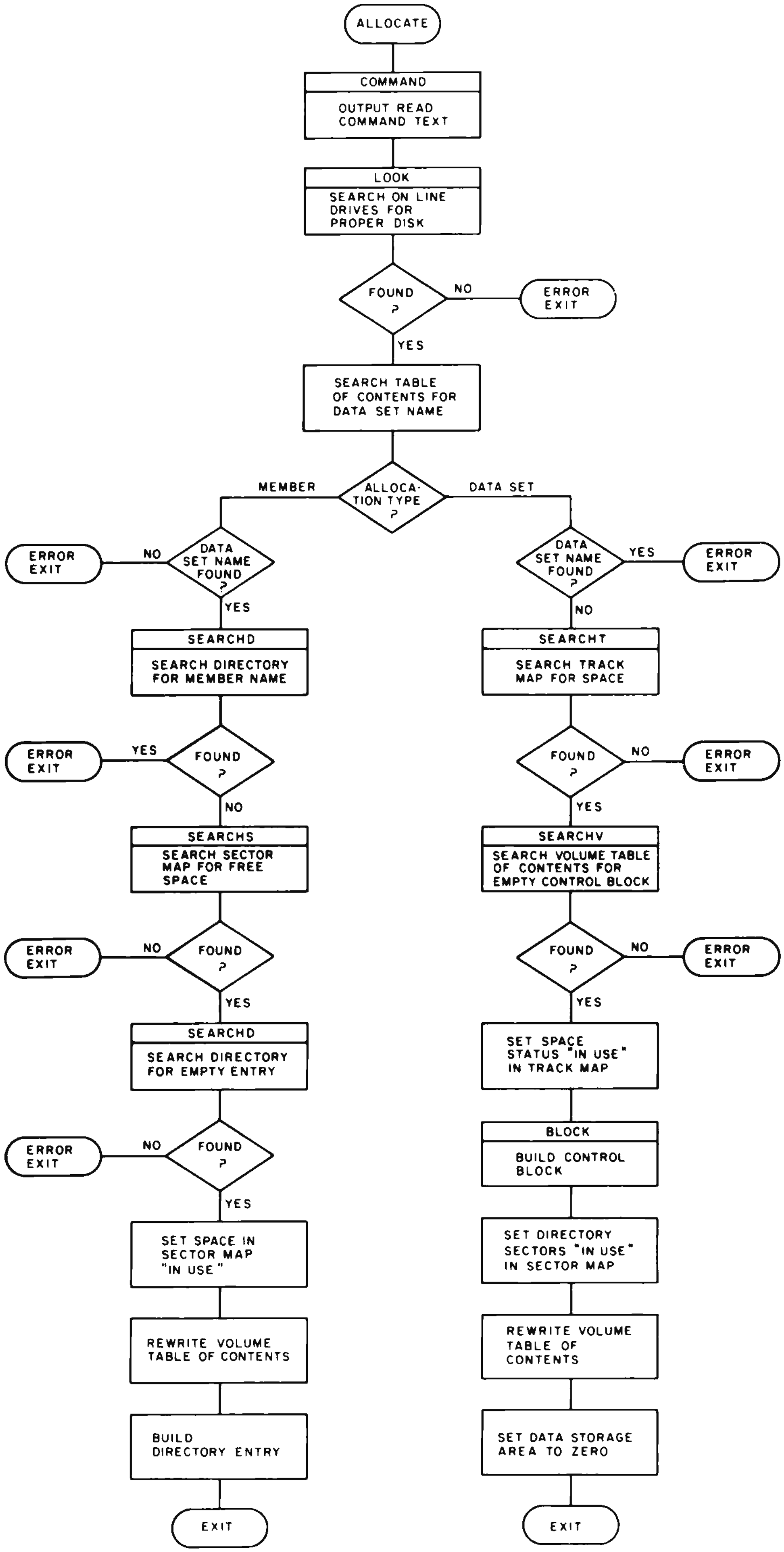


Figure 3: The ALLOCATE routine, used to put a new file onto the disk and update all pointers to indicate that the file is present.

are set to 0 (space used). The volume label, a count of the tracks used by the volume table of contents (permanently set to 1) and the maps are then written on the first sector of the first track of the disk. All other sectors of the table of contents track are set to binary 0. Every sector of the data storage area is also set to 0. Initialization of the disk is now complete.

Once the disk has been initialized, we can allocate data sets on it by using the allocate routine. The allocate routine shown in figure 3 actually consists of two routines: one to allocate data sets, and one to allocate members in a data set. To allocate a data set, the user enters the volume serial number of the disk, the name of the data set to be allocated (six characters maximum), the number of sectors to be used for directory, and the number of tracks to be devoted to the data area. Note that the space used for the directory is included in the amount of space entered for the data area definition.

The allocator routine then reads the volume table of contents track and verifies that the volume serial number on the disk matches that entered by the operator and that the data set name to be used does not already exist. If neither test fails, the track map is scanned for contiguous free space equal to the amount requested by the user. If the free space is found, the necessary bits are reset to 0 in the track map. The data set control block is now built by moving the required data into an empty block in the table of contents. The sector map is also updated to reflect the sectors used by the directory and the updated volume table of contents is rewritten on the disk, completing the allocation of a data set.

To allocate a member, the user must provide the volume serial number of the disk as well as the name of the data set of which the member will be a part, the member's name (six characters maximum), member data type, and the number of sectors to allocate to the member. The allocator program again verifies that the proper disk is on line and that the data set exists. Obtaining the address of the partitioned data set's directory from the control block for the data set, the allocator verifies that a member with the same name as that being allocated does not already exist. If all is well, the sector map is scanned for contiguous free space in the data set's data storage area sufficient to satisfy the user's request. If space is available, an empty directory record is used to create a directory entry for the member. The sector map is then updated and the directory and table of contents are rewritten on the disk.

We now have a partitioned data set with

a member allocated in it. To use this member, we need an access method to perform the input/output (IO) operations involved in opening the data set, finding the desired member, reading and writing the data in the member, and finally closing the data set. In addition, it is useful to have the access method open members sequentially. Given such an access method, we could graduate from doing physical IO to a far simpler logical IO in which we need only specify the volume serial number of a disk, a data set name and a member name in order to read or write data. The access method takes care of error processing, IO initialization and completion. The access method would be used by the source editor for saving source lines in members, and the assembler would use it for reading program source lines and writing program object code. This means that two members would be open simultaneously, but the access method should be able to handle that situation.

In order to free space on the floppy disk, deletion programs are required. Scratching a data set would deallocate the space used by the members in the data storage area (recorded in the sector map) and clear the track map bits used for the overall space allocation for the data set. The data set control block is then filled with binary 0s, freeing all space previously allocated.

Scratching a member requires that the sector map be updated and the directory entry for that member be rewritten as binary 0s. The deletion programs should be written to be as forgiving as possible of operator errors. After the operator has finished giving the delete command, the program should echo the command and wait for operator verification. All this is needed in order to avoid accidentally destroying irreplaceable data.

Another useful program is the volume table of contents lister. The list program reads the data set control blocks from disk, formats and displays the information contained in them, and indicates how much space is allocated to each data set and how much free space remains. As an added feature, the lister could be made smart enough to use the data in the data set control blocks to find and display the contents of the member's directory entries, thereby providing a powerful tool for controlling the data on the disk.

As with all good things, the free space on the disk will soon come to an end. The obvious solution is to delete a few members or data sets to make room for new things, but this has the disadvantage of destroying programs and data that may be important.

A better solution would be to place these programs into cheap archival storage in a format that simplifies their restoration onto the disk. Thus, the members or data sets to be saved would be written to magnetic tape by an unload program and written back on the disk later with a load program. The unload program reads the data set control block of the data set to be unloaded and writes it on tape. It then reads the directory entries for the members of the data set and writes the one selected to tape, followed by the data for the member from the data storage area. It continues to write directory entries followed by member data until the request entered by the user is completed.

The load program reads the data from the tape and if necessary allocates data sets and members before writing the data into them from the tape.

The partitioned data set file organization technique has been used successfully on many large systems and can be easily adapted for use by the hobbyist. In order to avoid a situation similar to that encountered with audio cassettes, standards should be formulated now so that disks can be interchanged from user to user. ■

Programming Quickies

Tic-Tac-Toe in BASIC

Mike Stoddard
16681 Lynn St
Huntington Beach CA 92649

Tic-Tac-Toe is a game that has been turned into a computer program thousands of times. Although it is fun to play with a computer, the best a human opponent can hope for is a draw game. After a short time the player loses interest because he or she never has any chance of winning. My version plays a regular game except that there are a few logic errors which put the player on a more even level with the computer. There is a fighting chance against the now imperfect opponent. The program logic that scans the rows, columns and diagonals has been altered to produce countermoves much like those of a person just learning the game. After playing dozens of games with the machine, if the player is lucky, the pattern of moves that fool the machine will emerge and allow the player to beat the machine every game. One such pattern is shown by the sample game of figure 1.

To change the program logic back to playing a perfect game every time, change the value of Z at line 1140 from 1 to 2. This hint should allow any good hacker to figure out how the program works internally. You can have fun playing the computer and showing your friends you are, indeed, smarter than the computer!

```
THE GAME OF TIC-TAC-TOE. THERE ARE 4 POSSIBLE GAMES
WHICH YOU CAN WIN. THE BOX SQUARES ARE NUMBERED:

     1  2  3
     4  5  6
     7  8  9

I AM NOW READY TO PLAY. YOUR MOVE FIRST.
? 4     I WILL PUT AN X IN BOX 5   YOUR TURN.
? 2     LET'S PUT AN X IN BOX 9   YOUR TURN.
? 1     AN X IN BOX 3 WILL STOP YOU.  YOUR TURN.
? 7     CONGRATULATIONS. YOU WIN THIS GAME.


    O  O  X
    O  X  -
    O  -  X

I AM NOW READY TO PLAY. YOUR MOVE FIRST.
?
```

Figure 1: This is one of the patterns that can be used to always beat the computer. There are three other winning strategies that will result in a win for the human player.

Listing 1: BASIC source code listing for Tic-TacToe game.

```
1000 REM "T4" ==== TIC-TAC-TOE WITH A TWIST
1010 REM 02-05-77  WRITTEN BY MIKE STODDARD
1020 REM
1030 DIM K(3,3),L(9),G$(12)
1040 PRINT "\215\THE GAME OF TIC-TAC-TOE.  THERE ARE 4 POSSIBLE GAMES"
1050 PRINT "WHICH YOU CAN WIN.  THE BOX SQUARES ARE NUMBERED:"
1060 PRINT "\215\     1  2  3\215\     4  5  6\215\  , 7  8  9\215\"
1070 FOR I=1 TO 3
1080   FOR J=1 TO 3
1090     LET K(I,J)=9
1100     LET L((I-1)*3+J)=0
1110   NEXT J
1120 NEXT I
1130 LET K3=1
1140 LET Z=1
1150 PRINT "I AM NOW READY TO PLAY.  YOUR MOVE FIRST."
1160 GOSUB  2980
1170 IF F=1 GOTO  1160
1180 LET L(1)=L1
1190 LET N2=1
1200 GOTO 2400
1210 IF K(2,2)<>9 GOTO  1250
1220 LET K(2,2)=4
1230 LET N=5
1240 GOTO 1270
1250 LET K(1,1)=4
1260 LET N=1
1270 LET L(2)=N
1280 PRINT  TAB (7);"I WILL PUT AN X IN BOX";N;"  YOUR TURN."
1290 GOSUB  2980
1300 IF F=1 GOTO  1290
1310 LET L(3)=L1
1320 LET N2=2
1330 GOTO 1200
1340 LET M=11
1350 LET N1=1
1360 GOTO 1890
1370 ON Z GOTO  1420, 1380
1380 IF L(1)+L(3)<>6 GOTO 1420
1390 LET N=1
1400 LET K(1,1)=4
1410 GOTO 1440
1420 LET M=22
1430 GOTO 1890
1440 LET L(4)=N
1450 PRINT  TAB (7);"LET'S PUT AN X IN BOX";N;"  YOUR TURN."
1460 GOSUB  2980
1470 IF F=1 GOTO  1460
1480 LET L(5)=L1
1490 ON Z GOTO  1570, 1500
1500 IF L1<>2 GOTO  1570
1510 IF L(1)+L(3)<>13 GOTO  1570
1520 IF K(1,3)<>9 GOTO  1570
1530 LET N=3
1540 LET K(1,3)=4
1550 LET K(1,2)=1
1560 GOTO 1760
1570 LET N2=3
1580 GOTO 1200
1590 LET M=3
1600 LET N1=5
1610 GOTO 1890
1620 LET M=17
1630 LET N1=2
1640 GOTO 1890
1650 LET M=11
1660 LET N1=3
1670 GOTO 1890
1680 LET M=27
1690 LET N1=4
1700 GOTO 1890
1710 LET L(6)=N
1720 GOTO 2960
1730 LET L(6)=N
1740 PRINT  TAB (7);"AN X IN BOX";N;"WILL STOP YOU.  YOUR TURN "
1750 GOTO 1820
1760 LET L(6)=N
1770 PRINT  TAB (7);"I WILL PUT AN X IN BOX";N;"  YOUR TURN."
1780 GOTO 1820
1790 LET M=27
1800 LET K3=5
1810 GOTO 1890
1820 GOSUB  2980
1830 IF F=1 GOTO  1820
1840 LET K1=3
1850 LET L(7)=L1
1860 LET N2=4
1870 GOTO 1200
1880 REM COLUUMN ADD SUBROUTINE
1890 LET K2=1
1900 FOR I=1 TO 3
1910   LET N4=0
1920   FOR J=1 TO 3
1930     LET N4=N4+K(I,J)
1940     IF N4=M GOTO 2230
1950   NEXT J
1960 NEXT I
1970 LET K2=2
1980 FOR J=1 TO 3
1990   LET N4=0
2000   FOR I=1 TO 3
2010     LET N4=N4+K(I,J)
2020     IF N4=M GOTO 2230
2030   NEXT I
2040 NEXT J
2050 LET K2=3
```

Listing 1 continued:

```
2060 LET N4=0
2070 FOR I=1 TO 3
2080   LET J=I
2090   LET N4=N4+K(I,I)
2100   IF N4=M GOTO 2230
2110 NEXT I
2120 LET K2=4
2130 LET N4=0
2140 FOR I=1 TO 3
2150   LET J=4-I
2160   LET N4=N4+K(I,J)
2170   IF N4=M GOTO 2230
2180 NEXT I
2190 IF K3=5 GOTO 2210
2200 ON N1 GOTO 1370, 1650, 1680, 1790, 1620
2210 PRINT  TAB (7);"THE GAME IS A DRAW."
2220 GOTO 2780
2230 IF M=3 GOTO  2260
2240 IF K(I,J)=9 GOTO 2500
2250 ON K2 GOTO 2290, 2310, 2330, 2360
2260 PRINT  TAB (7);"CONGRATULATIONS. YOU WIN THIS GAME."
2270 GOTO 2780
2280 REM COEFFICIENT EVALUATION SUBROUTINE
2290 LET J=J-1
2300 GOTO 2240
2310 LET I=I-1
2320 GOTO 2240
2330 LET I=I-1
2340 LET J=J-1
2350 GOTO 2240
2360 LET I=I-1
2370 LET J=J+1
2380 GOTO 2240
2390 REM CONVERT 1-9 SUBSCRIPT TO I-J VALUE
2400 LET N4=0
2410 FOR I=1 TO 3
2420   FOR J=1 TO 3
2430     LET N4=N4+1
2440     IF L1=N4 GOTO  2470
2450   NEXT J
2460 NEXT I
2470 LET K(I,J)=1
2480 ON N2 GOTO 1210, 1340, 1590, 1590, 2740
2490 REM CONVERT I-J VALUE TO 1-9 SUBSCRIPT
2500 IF N2<>4 GOTO  2550
2510 IF N1=2 GOTO 2540
2520 LET N1=5
2530 GOTO 2550
2540 LET N1=6
2550 LET K(I,J)=4
2560 LET N=0
2570 FOR J1=1 TO 3
2580   LET N=N+1 ■
```

Languages Forum

On Expressing Multiple Condition

David Faught
603 S Hazelton Av
Wheaton IL 60187

Having been interested and active in computer programming for some 10 years, I have watched with great anticipation the advent of the personal computer. In my own hobby and professional programming, I have used a rather large number of languages and have discovered through many painful experiences and uncounted hours of debugging that, in general, regardless of the language being used, a modular top-down approach to developing new programs is by far the easiest to understand and use. Unfortunately, my first experiences with programming consisted of occasional use of a Teletype terminal on a timesharing BASIC system. I still use BASIC for some of my hobby programs, but I find that unless some skill in program organization is used, a BASIC program can very easily become a rat's nest of inserted problem bypasses and altogether impossible to read. I recently obtained a copy of a primer on Pascal in the hopes that it would provide some knowledge and insight into providing a proper means of improving program structure through language format and syntax rather than relying on my own experience in this area. I can now see why this language has become much more popular as a first language in many universities and I hope that it will continue to grow in popularity and wide usage.

At various times in the past I have tried my own hand at designing a source language which would provide a much more meaningful approach to program structure which must be at least as important as function. I have no new language to propose in the cause of this interest, having never tried to implement one of the languages I have designed, but I do have some comments which may be of interest to those who are also involved in the search for the "perfect language."

In various languages, to my knowledge including BASIC, Pascal and COBOL, there is at least one statement which tests a condition and will or will not perform a specified function depending on the outcome of the test. This is, of course, the IF statement in the languages mentioned above. ELSE we forget, these languages also contain a statement which allows the testing of a variable state and the optional performance of one of several alternative functions depending on the state encountered. In BASIC, this is the "ON . . . GOTO . . ." statement; in Pascal it is the "**case** . . . **of** . . ." statement, and in COBOL it is the "GO TO . . . DEPENDING ON . . ." statement. FORTRAN also has this capability in a limited way through the use of the numeric IF rather than the logical IF. In my humble opinion, Pascal's implementation of this feature is far more meaningful not only to program structure but also to understanding the condition which is actually being tested. Many times when this structure is used in BASIC or COBOL, it is the powerful feature which justifies the use of that "hairy" computation to adjust the conditions which are actually present to be an integer between 1 and 10. Pascal's implementation of this structure is still not perfect because it takes some extra effort in defining data types to assure that one of the alternatives will indeed be picked. Correct me if I'm wrong, but there is no explicit way to specify what should be done if none of the states for which there are alternatives is actually

Figure 1.

```
IF     condition 1  statement 1
       condition 2  statement 2
       condition 3  statement 3

       . . .

       condition n  statement n
END
ELSE statement
```

Figure 2.

```
IF X
      <50 PRINT X;
      =50 PRINT X;X*X
      >50 PRINT X
END
...
IF X =
      1 SET Y=X
      2 SET Y, Z=X
      3 SET Z=X
END
ELSE
      SET X=1
...
IF X=0
      LET Y=Z*Z
   AND Y=1
      PRINT Z
   AND Y=2
      PRINT Z, Y
END
ELSE PRINT X
```

found. I believe it would also be somewhat tricky to use this single Pascal statement to perform one function if the variable being tested is less than 50, to perform another if the variable equals 50, or to perform a third if the value is greater than 50, for example.

These are obviously closely related conditions and would ideally be resolved with a simple statement structure. Note that these types of tests are possible in most any language: however, my suggestion is that there should be an alternative to this sometimes confusing structure.

Rather than having one statement to test a single condition and another to resolve multiple conditions, why not make the single condition test a simple subset of the multiple condition test? A loosely defined statement structure which would satisfy this requirement is shown in figure 1. As this statement is parsed, the statement becomes a multiple condition test when it contains multiple conditions. When additional conditions are encountered, each is concatenated to the first condition to form a new conditional expression which is then evaluated to determine if the statement associated with the new condition should be executed or not. The one restriction I would like to see on this type of a structure would entail not evaluating the original condition 1 if statement 1 is omitted. This means that only the concatenated expressions which are formed will actually be evaluated. Simple examples of possible forms of this type of a statement are shown in figure 2.

I believe that this statement structure provides an excellent aid to properly organizing program structure. It has the capability of directly relating associated states in an easy to understand manner and provides the flexibility which a multiple condition test should have, without having to go through any complex manipulations to resolve the conditions present to any particular restrictive form.

I would welcome any and all comments on this proposal and am always interested in finding more about the "perfect language" if you have any suggestions.

*Perhaps a reader with language design experience would care to comment on the various examples and suggestions proposed. Readers should note that none of the examples of figure 2 follow the prototype of figure 1 completely. But the examples of figure 2 might provide interesting variations on the multiple condition suggestion if they prove unambiguous to a language translator or interpreter. . . .*CH■

J O'Loughlin
908 Salem Dr
Huron OH 44839

Pascal Critique and a Comment

I have just finished reading your August 1978 issue and would like to comment. I am more than a little disappointed with the volume of coverage given to Pascal. Whatever the relative merits of the language that amount of discussion isn't merited in my opinion. There currently is no affordable implementation of the language available to the typical computer enthusiast. If and when Pascal is available I believe it will have a very rough battle trying to compete with both the pricing and heavy usage that BASIC enjoys today.

Another problem I think you have failed to address is the effect of the huge investment in time and money many have made in BASIC. Just what is to become of that? Conversion? An unlikely prospect given the historical example of the COBOL and PL/I controversy. Use both languages? Again an unlikely prospect. Most people have all they can handle without the demands a second language would require.

It should also be pointed out that Pascal has little or no following outside the academic community. It wouldn't be the first time that a language enjoying a great deal of admiration at the academic level has failed to gain acceptance as a viable tool in the real world of data processing. Languages are used and live only on the basis of perceived usefulness, the availability of experienced practitioners, and widespread implementation of the language. Pascal now has none of those attributes.

In my opinion the number of users of any language speak many silent volumes that by weight of numbers signal acceptance of a language more than any theoretical proposals or arguments about the relative advantages of competitive languages.

But there is an affordable implementation of Pascal — the UCSD system is available separately, or bundled with various manufacturers' products. As a means of learning a new language, conversion of one or two programs as tutorial experiments is just fine. Pascal should only be thought of as an avenue to more effective creation of new programs.

As for "no following outside the academic

Circle 386 on inquiry card.

community," Pascal has a very strong following as witnessed by the representatives of both industry and academia present at the UCSD Pascal Workshop last summer. BASIC was once the only logical and effective choice of languages to use.

*The virtue of Pascal and similar languages is the fact that the very expression of the program is so much closer to the way people think. I, for example, think in terms of "I want to do thus and so"; in Pascal, I might reference a procedure with the name thusandso. In BASIC I would have to reference it in the program with a number artificially created for that purpose. I might say GOSUB 10000, for example, when I really mean to call and execute a thusandso procedure. Pascal can be used as any other programming language — for the underlying computers are identical. It is a matter of making the expression of a program easier for the user. . . .*CH ■

Continuing Comments on APL

Timothy J Stryker
477 Hope St
Providence RI 02906

John Howland's "Comments on APL's Characteristics" in the May 1978 BYTE Languages Forum, page 143, are for the most part well thought out. However, it seems to me that he is missing something when he states that an APL programmer who composes programs on line is "similar to the person who opens his mouth and begins to speak before engaging his brain." The whole point of having an interactive language facility is based on the fact that the programmer does not always want to have to map out in advance exactly what it is he will ask the machine to do.

At the root of the matter is the consideration that there is no clean distinction between "implementing a program" and simply "invoking a system utility." Suppose, for example, we wish to check on the value of some variable, for example J, during an APL terminal session. We simply type:

```
J
```

Now suppose that the value of J is lower than we expected, so that we become interested in the first J elements of the array A. We type:

```
A[ι J]
```


But now perhaps we realize that J is actually the number of pairs of quantities in A and so in order to examine the set consisting of the first element of each pair we type:

$$A[{}^{-}1+2\times\iota\ J]$$

I could go on, but I think the point is clear: at what level of complexity have we stopped merely using the facilities of the environment and started writing a program?

My own view is that the above distinction is immaterial and that what matters is simply that the environment be structured to maximize the programmer's capability to accomplish the job at hand expediently. When the job at hand is large and complicated, there is no question but that at present hardware costs the most expedient recourse involves writing out the bulk of what is to be typed in beforehand. Likewise when the job at hand is trivial and transparent (eg: inspecting A[ι J], it is undeniably the case that writing it down before typing it in is a waste of time. In between these two extremes, things are not so clear, and the point at which paper and pencil become necessary will depend both on the individual programmer involved and on the system. However, what *is* clear is that the more complex the programmer finds himself able to get, on line, while still maintaining cogency of thought, the more productive he will be.

Mr Howland justly defends APL's right to left order of evaluation from those who would make it left to right: 3 ÷ 6 equals 2 is not a pleasant prospect. However, a valuable property for any language/environment to possess is one which allows short, transparent programs to be entered quickly and easily, without any need for pencil and paper in the process. My suggestion to the APL terminal manufacturers, if they haven't done so already, is to implement an option whereby each line could be *entered* from right to left (in much the way one frequently find oneself writing out lines of APL on paper, that is, starting at the righthand edge and working leftwards). In this way, the objective of simplified on line program creation could be achieved at no cost in the way of incompatibility with existing APL processing precedence.

Reacting just slightly to the last paragraph of your letter, a question comes to mind: is it the terminal manufacturer's responsibility or the APL interpreter-writer's responsibility to make the input sequence run from right to left? With a fast enough terminal, it is possible to rewrite the last *line with the new input in a nearly unnoticeable blink of the eye. Fast enough means a parallel memory mapped display, or a serial data terminal running at 9600 bps or higher. . . .*CH■

Calling Attention to HPL

Gerald Robb
5158 E Dwight
Fresno CA 93727

In response to the discussions of high level languages I have been following in BYTE, I would like to call your attention to an existing language as implemented on the Hewlett-Packard 9825A.

HPL, as Hewlett-Packard calls it, is implemented on the basic machine similarly to BASIC. Extensions are available by stages in read only memory. String capabilities are enhanced by a string ROM. IO handling by the general IO ROM is enhanced to be similar to FORTRAN. The advanced programming ROM extends the capabilities to cover, in large part, the characteristics of PIDGIN ALGOL as described in *The Design and Analysis of Computer Algorithms* by Aho, Hopcroft and Ullman (Addison Wesley). A couple of features should be noted. An assignment operator is used, allowing the equal sign to be only a relational operator. While the list of variable names is limited to single letters, the flexibility of substrings, local variables in subprograms and functions, and r variables provide for few problems in practical use.

I have been working with this system for a year and a half developing and implementing programs for an agricultural consulting firm. While the firmware, fully implemented, is probably close to 70 K, this is an excellent example of what can be done on a small system, and might just be a good system to emulate.■

Languages Forum is a feature which is intended as an interactive dialog about the design and implementation of languages for personal computing. Statements and opinions submitted to this forum can be on any subject relevant to its purpose of fostering discussion and communication among BYTE readers on the subject of languages. We ask that all correspondents supply their full names and addresses to be printed with their commentaries.

Technical Forum

A Proposal for a Kitchen Inventory System, or Don't Byte the Wand That Feeds You

Richard Shuford
Editor

Technical Forum is a feature intended as an interactive dialog on the technology of personal computing. The subject matter is open-ended, and the intent is to foster discussion and communication among readers of BYTE. We ask that all correspondents supply their full names and addresses to be printed with their commentaries.

A practical and natural application for your home computer is an inventory system for the kitchen. Such a system would relieve humans of the details involved in making out a grocery list.

One convenient way of keeping track of the various items in the pantry is to use the information that is now provided on most food packages specifically for that purpose, namely, the Universal Product Code, or UPC. This is, of course, put there for use by food stores, but there is no reason that the UPC cannot be used in the home.

The Universal Product Code appears on a product label as a patch containing bar codes, with a line of human readable numbers underneath. This distinctive design has now become familiar to most North American shoppers. Information contained in the bar codes can be read by an optical sensing device connected to a computer.

At present many computer experimenters are equipping their computers with an optical sensing device, a bar code reader that has a scanning wand, for the purpose of scanning the new machine readable software which uses similar bar codes. A good example of such software is the PAPERBYTE™ series of books which BYTE Publications produces. It is probable that the same scanning wand used for reading the software may be used to read the Universal Product Code. The scanning wand provides a quick and easy method to identify a given item without keystroking any information into the computer. It will, unfortunately, be necessary to do some keystroking to set up the system. One minimal implementation of an inventory system might be set up in the following manner. For each grocery item in stock, a data base would exist containing:

- a representation of the Universal Product Code for a given item,
- a human language description of the product including brand name, generic name, and size or quantity,
- the minimum quantity that should be kept on hand,
- optional information on the item's shelf life,
- any other information which is deemed useful (for example, which members of a household like a particular item).

As the supply of an item becomes depleted, the container is thrown out. Immediately before disposing of the container, though, the UPC bars are scanned. The computer stores this code in a table of items which are depleted and should be restocked. Prior to setting off on a shopping trip, the user requests a display (preferably on a hard copy device) of depleted products. The computer uses the depleted products table to reference the master data base, and retrieves the human language description of the product from the master data base and displays it for the user, along with other information. If the display is on a hard copy device, the user simply tears off the paper and uses it as a grocery list.

Probably the best way to establish and maintain the data base is with the use of an interactive program. The user would build the data base from scratch by starting with those items on hand at the time. For each item, the UPC bar codes are scanned, then the user types in the other information about the item. (Note that it is not necessary for the numeric product code to be keystroked by the user.) After the initial data is stored, the interactive program may be executed to update the file with information for new or different products. It is not necessary to start with a huge data base containing data for every possible product. Each household would keep information tailored specifically to its needs in its data base.

The size of the data base is dependent on the number of different products a given household buys. It should be noted that each brand of a given generic item has its own code. The data base may be kept small if a single brand of a given item is used consistently. The procedure of reading the UPC bar codes just before consigning the container to the garbage follows one of the cardinal rules of computer use, which is: *garbage in — garbage out.*

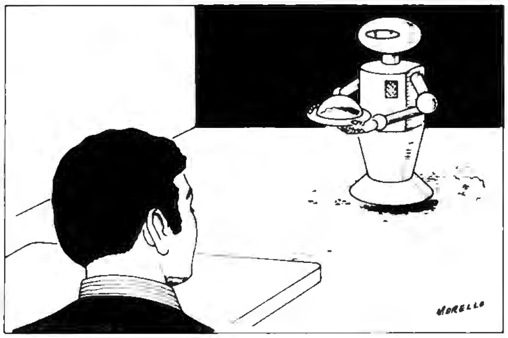
Special arrangements would still be necessary to handle those products not marked with the UPC in many stores, such as fresh meat and local produce. It might be possible, with knowledge of the encoding method of the UPC, to make a custom UPC bar code symbol by hand. This could be mounted permanently near the garbage can scanning station and scanned instead of a package symbol. The computer could then at least note that the supply of a nonencoded item was depleted and call attention to the fact.

An ingenious tinkerer could no doubt find many ways to improve the system. For example, some means to indicate exactly which nonencoded item is depleted might be devised. And it might prove useful to scan a package as it is bought, to verify that it is back in stock. Also, the addition of a modem for telephone communication gives rise to many possibilities.

A computer equipped with a modem could, with the proper programming, call a computer equipped food store and automatically order the necessary grocery items. And given the proper programming both in the home and at the store, it could dial up several different stores, compare prices and order from the store which provides the lowest price for the entire list. With electronic transfer of funds, the computer could even automatically pay the grocery bill.

Automating the kitchen inventory should give people the time to develop new recipes or new computer applications.■

He did not want his optional entertainment chip tampered with; his films, recordings, reading and fantasy trips had all been carefully selected, carefully tested over the years.



Lawrence F Willard
POB 202
Yalesville CT 06492

The Mother Chip

Jonathon Witherspoon Twombly floated up out of a warm and comfortable world of drifting, unconnected images to begin, unfortunately, a new day. He stared at the cream colored ceiling, as he always did, to read the wedge of light that fanned from the top of the window shade to intersect a discolored area of the ceiling in a fairly significant manner. Nine o'clock, he guessed, disdaining for three and a half seconds the absolute accuracy of his Minnie on the bedside table. He rolled over and looked at it, stubbornly translating 24 hour time to his own archaic measure. His guess was a mere four minutes and ten seconds slow. Pretty close, he noted contentedly.

The mini-mini-micro-processor, his own personal computer link to the vast and complex world of 1997, winked at him with a softly glowing numeral 3. Not much larger than a deck of playing cards and half as thick, Minnie rested upright in her umbilical slot, absorbing power for her batteries and sharing her thoughts with the house computer in its basement hideaway. A rather old microprocessor, Mike ran the house, but Minnie was boss as long she was plugged into the table, or as long she was on Twombly's person and not more than a mile distant from the house. Beyond that range she could integrate with Mike up to a distance of 40 miles using the high power car facilities, and beyond the 40 mile radius she could use the worldwide network of relay stations. But that cost money and was rarely necessary. Jonathon Twombly did not travel very much or very far.

At the moment he was staring at the glowing figure 3 in disbelief. Three messages for him during the night? How unusual! Twombly was a nobody; no family, no friends, no job. He didn't have to work, and so never would be allowed to. He lived on the regular income from his trust fund, and, with Minnie's help, he kept his outgo exactly equal to his income. It was a good life.

He removed Minnie from her niche and keyed in a command. The four foot square screen built into the wall at the foot of his bed lit up and the readout began:

MSG 1 BELLOGRAM 7.6.97 2207 66234621
BP P12 TWOMBLY, JONATHON W 779 28
88980 BMA
YOU ARE HEREBY CITED BY TRAFFIC
CONTROL OFFICER 229 BOSTON CITY
POLICE FOR MOVING TRAFFIC VIOLATION
7.6.97 1201 HOURS PL 2395 SEC. 8. B. SECTOR
QUADRANT 9 FINE 25 DOLLARS REMIT
WITHIN 24 HOURS TO AVOID ARREST
PC JOHN KELLEY.

Twombly swore mildly (he was not an aggressive man). He'd had the car on manual five minutes during the entire day and he'd managed to get a ticket. He might as well pay now and get it over with. He didn't even bother to call up the picture the cops had surely filed. He didn't want to see himself on the screen making an ass of himself. Payment of the fine would wipe the picture out of the police computer banks. He punched up his bank balance, confining it to Minnie's small screen. $207.81. Even with seven cents added as interest during the night, it wasn't a healthy balance. He swore again, mildly, and punched in the command, the amount and the police computer address, checked it on the screen and punched the execute button. $182.66. The city had $25; the bank had its 15 cent service charge. Twombly called up the second message:

About the Author

Lawrence F Willard has been a journalist and free lance writer for over 30 years, contributing to many magazines, including New Hampshire based Yankee Magazine. *A ham radio operator, Larry teaches journalism and media courses at Manchester Community College in Manchester CT.*

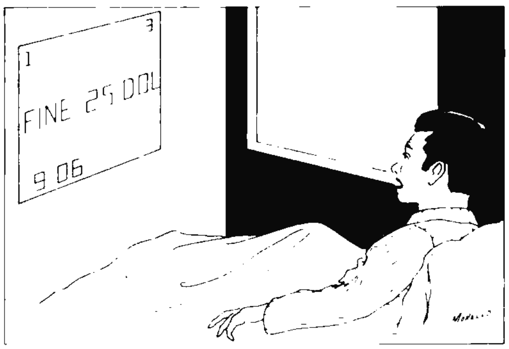
MSG 2 VIDEOPHONE CALL 0231 7.7.97
CENTRAL HOTEL RM 63
HI TWOMB OLD CHAP. REMEMBER ME?
PUDDY, ROOMY, WESTERN U? IN TOWN
FEW HOURS. HOW ABOUT A DRINK? NOW.
LEAVING SUNUP. 766 26 0589 CHEERS.

Twombly shuddered, wiped the screen clear. He had set Minnie to store night calls, not wake him up; he wasn't sorry. He called up the last message:

MSG 3 BELLOGRAM 7.7.97 0800 66091532
FBCC BOSSOFF TWOMBLY, JONATHON W
779 28 88980 BMA
YOU ARE HEREBY REMINDED
APPOINTMENT THIS OFFICE ANNUAL
CHECKUP AND FIVE YEAR
REPROGRAMMING 1400 HOURS THIS DATE.
PREPARE FOR ROUTINE PHYSICAL,
PSYCHOLOGICAL TESTS TO DETERMINE
AGING FACTOR. ALL NEW PROGRAM CHIPS
TO BE INSTALLED, INCLUDING OPTIONALS.
YOU MAY RETAIN OR CHANGE OPTIONALS.
NOTE: WHEN THE HELL ARE YOU GOING
TO LET US REPLACE YOUR ANTIQUATED
MINI FOR NEW ATOM POWERED MODEL?
P. T. HARRIS BUCHIEF.

Twombly wiped the screen. Of course he hadn't forgotten. Minnie had already placed an order for a car. Good old Minnie. Like hell they would replace her. Not yet. He sat for a moment, thinking. They'd find him five years older, reflexes a bit slower. New programming would compensate. Obstacle detection devices would take over a little sooner when he drove on manual, putting him a little less in control. His heart would be monitored more carefully and his medi-save unit would probably get newer, more powerful drugs. Minnie would probably calm him down a little quicker when he got overexcited. Did he want his optional entertainment chip tampered with? No. His films, his recordings, his reading, his fantasy trips had all been carefully selected, carefully tested over many years. They would do without futher change.

Twombly got up, dressed (he wanted no help from the waldos, those mechanical servants that Mike controlled), and dropped



Minnie into his inside jacket pocket from which nobody but he could remove it without calling out the police emergency vehicles and perhaps the National Guard; anyway, nobody had ever tried it. He went into his small but luxurious dining room to eat the late breakfast Minnie had summoned for him. He was served by one of his two household waldos, which glided across the carpet on a cushion of air, delicately bearing a plate of scrambled eggs in hands that could bend a steel I beam into a pretzel.

Mike operated the waldos, using their sensory substitutes for eyes, ears and sense of touch. Twombly had programmed Mike to speak through the waldos' vocoder systems in an almost human male voice, and he had also fixed it so that when Minnie wanted to speak to him verbally through Mike, the waldos underwent a startling change of sex, answering in suitable feminine tones. She spoke to him now above the plate of scrambled eggs.

"I have ordered you a car, a Whinger Electric, to be here at one o'clock. Very few of the agencies still have those in their inventory. They are, as you know, obsolete. Next year we will have to pay the antique car premium to get one."

"I know, I know; we'll worry about that next year." Twombly ate his eggs, retired to his study and programmed the next lesson in his study course, "Late 19th Century and Early Twentieth Century External Bathroom Architecture in Rural Areas." Fifteen minutes before the hour of one o'clock he was standing on the sidewalk in front of his townhouse awaiting the Whinger Electric. He was not alone. Standing a few feet away from him was the occupant of the neighboring townhouse, a Professor John Carmody, who taught English to first year students at Radcliffe. Twombly wondered whether he should activate his nonintercourse signal, but Minnie's low buzz indicated that the good professor had activated his own, thus solving the problem. It was, of course, the grossest kind of social blunder to speak to or take any notice of a person radiating a nonintercourse signal.

Just then the signal stopped, and before Twombly could activate his own signal, the professor spoke to him.

"Since your signal is off, sir, may I be permitted conversation with you?"

It was now too late to emit a nonintercourse signal, and the grossest kind of social impropriety not to answer.

"My dear Professor Carmody, you may indeed converse with me," said Twombly. "Nice day, isn't it?"

"Yes, I perceive you are waiting for a car?"

"Indeed, yes. I am headed into the center of the city, to government sector. Do you wish to share my car?"

"No, no, my dear Mr Twombly; my own is on the way. Would you be interested in a small wager, say five dollars, as to which vehicle arrives first?"

"That would be most sporting," agreed Twombly. "Shall we say ten dollars?"

"Done," said Professor Carmody.

Although neither could now erect the nonintercourse barrier, by mutual unspoken agreement nothing more was said. At exactly one o'clock both vehicles came into view, arriving from opposite directions. Twombly's arrived a fraction of a second before the professor's. The professor nodded in token of defeat, and entered his car. Twombly's bank balance would shortly grow by ten dollars. He felt very good about that. Entering the two seater electric, he took Minnie from his inside jacket pocket and inserted it in the slot in the dashboard. It was now his car, for a daily rental fee, until he removed Minnie and gave the car a signal to return to its depot. He put the car on automatic and keyed in the destination. He could not get a traffic ticket as long as the car was under automatic control by the city's own traffic computer which directed the symbiotic duo of Minnie and the car's computer.

Twombly leaned back, completely relaxed in the knowledge that he was in the safest environment ever known to mankind. No matter what difficulties there were, through rain, fog, sleet or snow the car would transport him without danger. If he had a heart attack, his medisave implant would go into action, administering adrenalin, electric shock, or whatever else was needed for the few minutes it would take for help to arrive. Minnie would work through the car computer and signal system to coordinate the meeting of the car with the nearest mobile medical unit, which would be receiving a flow of medical data and electrocardiograms. It was exceedingly difficult to die in an automobile, or on the street for that matter. Minnies could work directly into repeaters mounted on telephone poles no more than a mile apart throughout the entire city.

At government center the car parked itself to wait until Twombly's return, since he had not given it instructions to return to the depot. He took Minnie out of the dashboard slot and returned it to his inside jacket pocket, stepped out of the car onto a moving walkway, and was carried into the building that housed the Boston office of the Federal Bureau of Computer Control. He took the elevator to the twelfth floor

Office of Programming and Adjustment, where he underwent a battery of tests which proved that he was five years older. His Minnie was sent to one of many laboratories where highly skilled technicians made new program chips and inserted the chips in the Minnie to replace the ones which had served Twombly well for five years. It was late afternoon when Twombly left; an hour after that, one of the technicians approached the lab chief with an almost microscopic program chip in the palm of his hand.

"We have a condition red, I think," he told the chief. "This is the alternate program entertainment chip from Twombly's Minnie."

"Carson," said the chief, "that simply cannot be. He couldn't get out of the building without a full complement of chips; the master computer wouldn't let him through the door."

Carson, his face almost as red as the little dot on the chip which meant alternate program, said: "He had a full complement of chips. I got the wrong one in. He got an experimental chip I was designing for my wife's Minnie."

"What kind of an experimental chip?" asked the chief in tones that made Carson's flesh creep.

"You might call it a babysitting chip," said the technician, "although it doesn't just sit. I can tell you that we're in a great deal of trouble if he activates that chip. We have to prevent that."

"Condition red," sighed the chief. "We have to key into his Minnie by way of the house computer, but we'll have to get authorization from Washington. I'll notify Harris; it's his problem. He won't like it much."

"I don't think we have time. He'll most likely activate the entertainment chip after he finishes dinner; Twombly is predictable."

"We have to take time. After that J E Lewyt scandal, where the untouchability of our beloved director was found wanting, we've been under very rigid orders about invading the privacy of private computers. We've got to get authorization."

They got it after a three hour delay, but as Carson feared, it was too late. When the special code got them access to the Twombly house computer, it reported that Twombly had activated the alternate program entertainment chip. The chief sighed and requested a complete readout from the time of activation.

CHIP ACTIVATED 2030 HOURS. SEQUENCE COMPLETED: UNDRESSING, BATHING, DRYING, POWDERING, DIAPERING. AS INSTRUCTED BABY HAS BEEN PUT TO BED

WITH WARM BOTTLE. BURPING TO TAKE PLACE ON COMPLETION OF FEEDING. ESTIMATED TIME: THREE MINUTES FROM NOW.

"We can do without the burping," yelled the chief. "Carson, override the program at once; switch off the alternate program. My God, I think we have a law suit on our hands. You and I will end up in the coal mines."

"Program is off, chief. I'll see if I can get an informal but detailed report from his Minnie. . . it's coming now."

Chief of Laboratory Q, George Justine, had chewed the nails down to the quick on one hand and had started on the other when Carson leaned back in his chair and actually smiled.

"Twombly started to panic when the waldos grabbed him and started to undress him, but calmed down and gave in when he couldn't stop them. He seemed to be actually enjoying the bath, and when he was put to bed with the warm bottle he slurped it down and actually cooed. He is now in a deep, peaceful sleep, and his Minnie reports that his blood pressure is normal for the first time in months."

"Well, we're not off the hook yet, but it looks better."

"Chief, we'll call him in tomorrow and explain the mistake, and apologize. We'll give him back his entertainment chip and I'll take back the babysitter chip."

"I doubt it. I mean, we can call him in, but something tells me he isn't going to give up that chip. It fits in too well with his psychological profile. We'll have to give it to him in addition to the entertainment chip. We'll gain one thing; I think we can get him to take the newer model Minnie, because the one he has doesn't have room for any more alternate programs. If he wants to play baby, he'll have to exchange Minnies, and I think he will."

"I hate to lose that babysitter chip; I put a lot of work in on that."

"Carson, that's going to be the least of your worries. We're going to have to fill out lots of reports. . . *you* are. There'll be lots of investigations and an awful lot of flack. There is one possible ray of light: there may be other people like Twombly, and this may prove to be some kind of legitimate therapy. I don't know. That's for the psychologists to decide. Right now we have to get ready for the worst, charges of invasion of Twombly's privacy. We panicked. We went to the top to get authorization to enter the computer of a private citizen, citing clear and present danger. What did we achieve? We stopped a man from getting burped."■

Desk Top Wonders

Commander in Chief

A Game for the TI-58 Programmable Calculator

Larry Kollar
Room 225 W Wadsworth Hall
Michigan Technological University
Houghton MI 49931

Commander in Chief is a TI-58 snowball war game for one player (see listing 1). After entering the program, press E. This clears the memory and initializes the random number generator (program 15 in the library module). You are now ready to play. Enter the number of snowballs you want up to 100 and push A. If you try to enter more than 100 snowballs, the program will place only 100 snowballs in your register. No iceballs allowed.

After a few seconds, the calculator will come back with a 1 or a flashing 1. If the display flashes, you are at war. Next, you estimate how many snowballs the enemy has and push B. The display will flash how many snowballs the enemy actually has. Following this, it will display a 1, 0 or –1 and then the year number; or the display will flash 9.9999999 99. If this occurs, there has been a holocaust and the enemy is rendered inoperative. If there is no holocaust, the 1, 0 or –1 tells you whether you have won, achieved a standoff, or lost; then the year number is displayed. You and your enemy have lost half of your snowballs and each of you will add more on the next year.

If there is no war at all during the year, you have the option of declaring war. The procedure is the same as that in which the enemy has declared war. If you can make it through ten years, you win the Snobel Peace Prize.■

ACKNOWLEDGEMENTS

"Commander in Chief," *The Pocket Calculator Game Book,* Schlossberg and Brockman, Bantam Books, 1976.

Thanks to David Nahakian for helping me with some of the program sequences.

Sample Game

Year	Your Total Snowballs	Total Enemy Snowballs (not seen)
1	100	68
(No war, go on to the next year)		
2	180	127
(No war)		
3	260	209

(Enemy declares war. Player estimates 191 snowballs, an error of 18. This is multiplied by the actual enemy snowballs and the number of his snowballs. The resulting holocaust factor is 978,120. The holocaust factor needed to cause a holocaust is 1,500,000. There has been no holocaust, so each power loses half his/her snowballs, discarding fractions. Player wins.)

Year	Your Total Snowballs	Total Enemy Snowballs (not seen)
4	130	104
(No war)		
5	230	197
(No war)		
6	330	224

(Player declares war and estimates 251 enemy snowballs, an error of 27. The holocaust factor is 1,995,840. There has been a holocaust; and both sides are blitzed.

Loc.	Keys						Commentary
000	*Lbl	E	*CM's	*Pgm	15	E	Random number generator.
006	*Pgm	15	*E'	R/S	*Lbl	A	
012	x⇄t	1	0	0	x ≥ t	–	100 is maximum number of snowballs
018	x⇄t	*Lbl	–	x⇄t	SUM	00	added is one turn.
024	1	STO	10				Lower limit
027	1	0	1	STO	11		Upper limit
032	*Pgm	15	C	*Int	SUM	13	Add enemy snowballs
038	1	3	STO	11			
042	15	C	STO	20			
046	9	x⇄t	RCL	20	*x ≥ t		War ?
051	*x̄	*Lbl	+	1	SUM	21	
057	RCL	21	R/S	*Lbl	B	–	
063	RCL	13	*Pause	=	|x|		Enemy snowballs flash
068	(	CE	X	RCL	00	X	Holocaust factor
		RCL	13	)			
077	STO	15					
079	1	5	0	0	0	0	Maximum holocaust factor
		0					
086	x⇄t	RCL	15	*x ≥ t	lnx	RCL	
092	13	–	2	=	*Int	STO	
098	13	RCL	0	÷	2	=	
104	*Int	STO	0	–	RCL	13	
110	=						
111	*Op	10					Signum function (Who won ?)
113	*Pause	Pause	RCL	21	GTO	+	
119	*Lbl	*x̄	CLR	÷	=	RCL	Flash year number
		21	R/S				
127	*Lbl	Lnx	CLR	1/x	*CM's	R/S	You are rendered inoperative!

Listing 1: Commander in Chief, a game for the Texas Instruments TI-58 programmable calculator. Note that some of the operations of this TI calculator series allow multikey entries into one location. This is indicated by an asterisk within the key.

FORTRAN and Its Generalizations

W Douglas Maurer
Room 634
University Library Building
George Washington University
Washington DC 20052

It really is a necessary part of your knowledge, even if you're never going to write FORTRAN programs.

The average small system user will not be able to use FORTRAN as a programming language for his or her system. Some small systems have BASIC, and there is at least one FORTRAN system for the 8080 (available from Microsoft), but FORTRAN is still chiefly a language for large computers (including minicomputers and mid-size computers). Nevertheless, there are many situations in which a knowledge of FORTRAN is important even to the small system user. The most important of these is in the description of algorithms. It is of no use to describe an algorithm in, say, INTEL 8080 assembly language, since this would not make sense to users of Motorola and other microcomputers; so algorithms are very often described either in FORTRAN, or in some other algebraic language. FORTRAN, though, seems to be the one that is used most often for this purpose, since more people know FORTRAN than any other algebraic language.

There are hundreds of books on FORTRAN today, all of which are written for the large system user who is, presumably, actually going to use FORTRAN to solve problems. It is very rare that one finds a description of FORTRAN written for those who merely need to understand algorithms written in FORTRAN, but who are going to rewrite those algorithms in some other language themselves. The present article is written to fill this need.

The basic function of an algebraic language, of course, is to allow one to write algebraic expressions directly. Given a formula like

$$k = \frac{ij-i+j}{n}$$

one has to write, in assembly language, something like "load i; multiply by j; subtract i; add j; divide by n; store in k" in order to calculate the new value of k. On most small systems, the job is even harder than this. We have to call subroutines for multiplication and division, and in an 8080 based system, even addition and subtraction of quantities in memory cannot be done directly: the right addresses have to be loaded into H and L first. However, when we are describing an algorithm, rather than writing a program, the formula above is what interests us, and we would like to write it directly. In FORTRAN, we would write

```
K = (I*J–I+J)/N
```

There are several differences between the FORTRAN version and the original formula. Some of them are due to the fact that we have to be able to key the FORTRAN formula into a system on a terminal or a keypunch. For instance, we have to use upper case letters instead of lower case and we have to use the slash (/) to mean "divide." The parentheses are necessary because, if we did not use them, that is, if we wrote

```
K = I*J–I+J/N
```

the formula we would be expressing would actually be

$$k = ij-i+\frac{j}{n}$$

since division takes precedence over addition.

The last difference between the formula and its FORTRAN version is in the use of the asterisk (*). This is necessary whenever we have a multiplication, since **IJ**, just as in assembly languages, would be the name of a single variable. In FORTRAN, the name of a variable must start with a letter, can contain only letters and digits (although some versions of FORTRAN allow a few extra characters, most do not), and has a maximum length which depends on the system being used. Typical maximum lengths for identifiers are eight characters (IBM 360 and 370) and six characters (UNIVAC 1100 series, DECsystem 10).

In addition to the use of formulas of this kind, FORTRAN involves a number of other statements which express commonly encountered sequences of instructions. Among these are:

(1) **GO TO**. Where the 8080 assembly language user writes **JMP K**, meaning "Jump to K," and the 6800 user writes BRA K, meaning "Branch to K," the FORTRAN user writes **GO TO 15**, meaning "Go to statement number 15." Statements in FORTRAN have numbers rather than names, and the numbers have nothing to do with addresses in the machine; they can be assigned arbitrarily and do not even have to be in sequence (as they do in BASIC).

(2) **IF**. The keypunches used by many large system users do not have the characters <, >, ⩽, ⩾, or ≠ (although they do have =) and FORTRAN therefore uses **.LT.** (less than), **.GT.** (greater than), **.LE.** (less than or equal), **.GE.** (greater than or equal), and **.NE.** (unequal). Thus "If A is less than B, then go to statement number 15" would be written in FORTRAN as

IF (A.LT.B) GO TO 15

FORTRAN is distinguished from BASIC (and ALGOL, PL/I, and various other algebraic languages) by requiring the parentheses after the keyword **IF**, and also by not making use of the word **THEN**. FORTRAN also uses **.EQ.** (equal) in comparing, and not the character =, which is reserved for assignment statements involving formulas (such as in **K = (I*J−I+J)/N**, discussed above).

(3) **STOP**. This signals the end of an algorithm, although a large system will not actually stop at this statement, but will go on to do the next job (assuming that there are more jobs waiting to be done).

(4) **END**. This is simply the last statement in a program and has nothing to do with stopping, which can happen at any time. That is, we can have several **STOP** statements in a program, but only one **END** statement.

(5) **READ**. A **READ** statement in FORTRAN is largely self-explanatory; thus **READ (5, 91) N, A, B** reads in three quantities and calls them **N**, **A**, and **B**. The **5** in this statement is a FORTRAN convention: the standard input medium (as opposed to any special tapes or disk files which might be used) is referred to as unit number 5. The **91** is a reference to a **FORMAT** statement which describes, in this case, in what format **N**, **A**, and **B** are going to be given. This **FORMAT** statement can be ignored by the person who is merely interested in what the algorithm does.

(6) **WRITE**. This is very much like **READ**, except for one peculiar convention: when one of the quantities to be written out is a constant string, then this string is found in the associated **FORMAT** statement, rather than in the **WRITE** statement itself. An example should make this clear. Suppose we want to write out the sentence THERE ARE 7 ERRORS IN THE ABOVE PROGRAM. We have a count in our program

called NERRS, which is, in this case, equal to 7. We would like to write a statement something like

```
WRITE "THERE ARE ", NERRS,
 " ERRORS IN THE ABOVE PROGRAM".
```

In FORTRAN, however, we have to write something like

```
WRITE (6, 92) NERRS
```

where statement number **92**, the **FORMAT** statement, is

```
92  FORMAT (' THERE ARE ', I3,
 ' ERRORS IN THE ABOVE PROGRAM').
```

The **I3** here is the format for **NERRS** (a three digit integer), while the **6** in the **WRITE** statement is like the **5** in the preceding **READ** statement; that is, unit number 6 is the standard output medium.

Where constant strings are not present, **WRITE** is very much like **READ.** That is, we can have a statement **WRITE (6, 93) N,A,B** which will write out the quantities N, **A**, and **B.** In some versions of FORTRAN, we find **PRINT 93,N,A,B** with the unit number 6 left out; the only thing to remember here is that we are not printing out the number 93, as this is the **FORMAT** statement number, just as before.

(7) **DO.** Suppose we want to repeat a group of statements **N** times. Then, just before these statements, we can write

```
DO 25 I = 1, N
```

where the last statement in the group to be repeated has statement number **25.** This will not only cause the statements to be repeated, but will set the index **I** to a different value each time: 1 the first time, 2 the second time, and so on up through **N** the last time.

(8) **CALL.** The FORTRAN programmer can write **CALL SUB**, just like the 8080 programmer (the 6800 programmer would write **JSR SUB**, meaning "Jump to subroutine **SUB**"); the difference arises when the subroutine **SUB** has parameters. Where the small system user has to figure out his own way of passing parameters, FORTRAN does this automatically. If the parameters are **A, B,** and **C**, for example, the FORTRAN programmer simply writes **CALL SUB(A, B, C).**

(9) **SUBROUTINE.** At the beginning of every subroutine there is a statement like **SUBROUTINE SUB(X, Y, Z)**, which says that the name of this subroutine is **SUB** and that its dummy parameters (sometimes called formal parameters) are **X, Y,** and **Z.** This means that if **SUB** is now called as above (that is, with the statement **CALL SUB(A, B, C)**), then **X** corresponds to **A,**

Y corresponds to B, and Z corresponds to C.

(10) **RETURN**. This is used in a subroutine in place of **STOP**; it stops the subroutine and returns to the program (which could possibly be another subroutine) which called this subroutine. If we use **STOP** in a subroutine, the entire job will stop.

(11) **FUNCTION**. In FORTRAN there are certain special functions: **SQRT** (square root), **SIN** (sine), **COS** (cosine), and the like. Thus the FORTRAN statement **Y = SQRT(A)** sets **Y** equal to the square root of **A**. But FORTRAN also allows the programmer to make up his own functions. These are coded like subroutines, with two exceptions. We start a function with a statement like **FUNCTION F(X, Y, Z)** which tells us that F is the name of the function and **X**, **Y**, and **Z** are the dummy parameters. At the end of the function (normally just before **RETURN**) we write **F** = *e*, where *e* stands for whatever we want the value of the function to be. If we then use the function **F** by writing **U = F(A, B, C)**, then, just as before, **X** corresponds to **A**, **Y** corresponds to **B**, **Z** corresponds to **C**, and *e* will now be computed and **U** will be set equal to *e*.

(12) **DIMENSION**. This is used to define tables (arrays). **DIMENSION A(50)**, for example, defines a table of 50 variables which are called **A(1)**, **A(2)**, and so on up through **A(50)**. We can also, of course, make reference to **A(I)**, **A(J+1)**, and the like. **DIMENSION B(3, 3)** defines a matrix of nine variables, **B(1, 1)** through **B(3, 3)**, and we can make reference to **B(I, J)** if **I** and **J** have values 1, 2, or 3.

(13) **REAL**. Most large systems, of course, have floating point representations for real numbers. FORTRAN assumes that *every* variable represents a real number unless its name begins with I, J, K, L, M, or N, and even this rule can be overridden by a **REAL** statement. Thus **REAL LAMBDA** specifies **LAMBDA** to represent a real number, even though its name begins with L.

(14) **INTEGER**. The **INTEGER** statement allows us to define variables whose names do not begin with I, J, K, L, M, or N to be integers rather than real numbers. An integer on a large system is typically 32, 36, 48, or 60 bits long; an integer on a minicomputer or a midsize computer is typically 12, 16, 18, or 24 bits long.

(15) **COMMON**. Normally, when we have a variable in a subroutine which is called (for example) **J**, and another variable in the main program (or another subroutine) which is also called **J**, these are treated by FORTRAN as two different variables. The

exception to this rule occurs when J appears in COMMON statements in *both* programs. The rules for writing COMMON statements properly are complex; but in a published program, one may always assume that the rules have been properly followed.

Every so often, one will be faced with a program written in some algebraic language other than FORTRAN, such as ALGOL or PL/I. The main differences between these languages are as follows:

(1) **GO TO**. Statements in FORTRAN and BASIC have numbers, but statements in ALGOL and PL/I have names. When a name is defined it is followed by a colon.

(2) **IF**. Most algebraic languages other than FORTRAN use the additional keyword **THEN**, and many also allow the keyword **ELSE** (meaning "otherwise"). Thus **IF** α **THEN** β **ELSE** γ means "If α is true, then do the statement(s) β; otherwise, do the statement(s) γ."

(3) **STOP**. ALGOL does not have a **STOP** statement; to stop in the middle of a program, one writes **GO TO** α, where α is a label (followed by a colon) just before **END** at the end of a program.

(4) **END**. In ALGOL and PL/I there are two kinds of **END**. One is used just as in FORTRAN, and the other is in the middle of a program paired with **BEGIN**. The statements between **BEGIN** and **END** are called a block (or sometimes a compound statement), and may take the place of a single statement wherever one can legally occur in the language. PL/I also requires an **END** paired with each **DO**.

(5) **READ**. PL/I has two kinds of **READ**, one called **READ** and the other called **GET**. The **GET** variation is used when built-in format conversions are to be exercised. Some variations of **GET** involve no IO at all, causing conversions in memory. ALGOL, as it was originally defined, has no input statements at all, but many ALGOL programmers assume that there is a subroutine called **inreal(x)**, which inputs the real number **x**, and similarly **ininteger(x)**, which inputs the integer **x**.

(6) **WRITE**. PL/I uses WRITE as well as another form called PUT. **WRITE** corresponds to **READ** and **PUT** corresponds to **GET**. ALGOL has **outreal(x)** and **outinteger(x)** to correspond to **inreal(x)** and **ininteger(x)**.

(7) **DO**. In PL/I, in order to repeat certain statements from I = 1 to N, we write **DO I = 1 TO N** (note the word **TO**), followed by the statements to be executed, followed by **END**. In ALGOL, we write

for I:=1 step 1 until N do begin, followed by the statements, followed by end; if there is only one statement to be repeated, then begin and end are not necessary (although they may appear). In BASIC, we write FOR I=1 TO N, followed by the statements, followed by NEXT I.

(8) CALL. In BASIC we write GOSUB *n*, meaning "Go to a subroutine at statement number *n*"; subroutines in BASIC do not have names as they do in FORTRAN, ALGOL, and PL/I. In ALGOL, we leave out the word CALL; thus SUB(A, B, C) by itself is a statement which calls the subroutine SUB.

(9) SUBROUTINE. Subroutines in ALGOL and PL/I are called procedures, and where in FORTRAN one would write SUBROUTINE SUB(X, Y, Z), in ALGOL one writes procedure SUB(X, Y, Z), and in PL/I one writes SUB: PROCEDURE(X, Y, Z). The situation in ALGOL is especially confusing because a subroutine is written *inside* the program of which it is a subroutine, at the beginning of that program with all the other declarations (real, integer, and the like). This makes it very difficult, in practice, to figure out where the first statement of an ALGOL main program is, particularly if it has a lot of nested subroutines. You have to start at the beginning of the program and work your way through all the subroutines, each of which is declared by a procedure statement with a matching end (which you have to find); then you suddenly come, with no warning, upon a simple statement like i:=1 and, believe it or not, that is where you are supposed to start executing.

(10) RETURN. In PL/I you write RETURN(*e*) to correspond to F=*e* followed by RETURN in FORTRAN, where F is the name of a function.

(11) FUNCTION. The terms corresponding to the FORTRAN FUNCTION for ALGOL and PL/I are INTEGER PROCEDURE, REAL PROCEDURE, and the like; the adjective before PROCEDURE tells you whether the value of the function is supposed to be an integer, a real number, or whatever.

(12) DIMENSION. In BASIC, one writes DIM instead of DIMENSION. In ALGOL, one writes integer array or real array; in PL/I, one writes DECLARE, which may be shortened to DCL (and usually is). DECLARE in PL/I is an all-purpose declaration having dozens of variations, but DECLARE A(100), sometimes followed by various other keywords, is roughly like DIMENSION A(100) in FORTRAN, as is real array

A[1:100] in ALGOL (the 1 here is the lower bound on subscripts, which may be arbitrary in ALGOL, although it is always 1 in FORTRAN).

(13) **REAL.** In ALGOL, the **REAL** attribute refers to representation as a floating point number. *[Note that the attribute* **FLOAT** *performs this function in PL/I, and that* **REAL** *in PL/I is used only to distinguish real from complex numbers. . .***BL***]*

(14) **INTEGER.** BASIC assumes that all numbers are real; integers will be treated as if they are real numbers, which usually works the way we want it to, although some operations like division must be watched carefully. In ALGOL, all integers must appear in **integer** statements.

(15) **COMMON.** In PL/I, all main routine variables are common (called "global" in PL/I parlance) to internal subroutines (ie: the subroutine is declared by a PROCEDURE statement within the boundaries of the calling PROCEDURE and its END) unless it is redefined in the subroutine. The EXTERNAL attribute is used to share variables between external procedures. In ALGOL, any variable in a main program may automatically be used in any of its subroutines, unless there is another variable declared in the given subroutine that has the same name.

(16) **Assignment statements.** In ALGOL, the symbol := is used where = is used in FORTRAN, BASIC, and PL/I. In addition, = is used where **.EQ.** is used in FORTRAN. Some versions of BASIC permit, and some require, the word **LET** at the beginning of every assignment statement.

(17) **Semicolons.** Every statement in ALGOL ends with a semicolon unless it is followed by **end**. Every PL/I statement is followed by a semicolon.

There are hundreds of other differences between the various algebraic languages, but these are the basic ones which are required to be able to read published algorithms in FORTRAN, ALGOL, BASIC, and PL/I. Most such algorithms, with a few notorious exceptions, are presented in such a way as to use only the rules described above. The reader whose appetite has been stimulated by the possibilities of algebraic languages might do well to supplement his small system knowledge by renting a small amount of time (perhaps $100 worth) on a large system and trying out various features of FORTRAN, PL/I, and the like. This is, of course, in addition to the use of cross assemblers and cross compilers, which still require large systems to produce small system object code.■

Technical Forum

Comments on the RF Entry Method for Video Monitors

Victor A Wiseman
7960 Grand Oaks Ct
Gurnee IL 60031

Photo 1: Author's system with Processor Technology SOL-20 computer and rear view of RF entry video display.

Photo 2: Typical display of the system.

This is a reply to a statement in Mr Fylstra's article "Convert Your TV Set to a Video Monitor," which appeared in the May 1978 BYTE, page 22.

While I will not contest Mr Fylstra's statements and arguments that the direct video entry method is definitely superior to the radio frequency (RF) entry method, I must take a stand against his statement that the radio frequency entry method ". . .is enough to display at most about 32 characters per line." Mr Fylstra has accurately identified and reported the pros and cons of the two methods, but an individual reading his article and contemplating a 64 or 80 character per line display would immediately discard the possibility of the radio frequency entry method. My experiences should prove this to be unfair.

When considering the options I had for adding a video display to my SOL-20, I considered buying a monitor for $180, converting my television for direct entry, and using the radio frequency entry method. Since I already had a portable television suitable for the job, I decided against spending $180 for a monitor. This left me with the direct video and radio frequency entry methods. I then armed myself with a Sams *Photo-Fact* folder and performed some exploratory surgery on my television. This convinced me that I could use the direct entry method, but it would require some care and time to do properly. I finally decided that the most expedient method would be the radio frequency entry method; the cost was low enough so that, if it didn't work out, I would not have lost much. As it turned out, the radio frequency entry method proved entirely satisfactory for my needs and I have been using it for the past year and a half.

Photo 1 shows my system. The processor is a SOL-20 which incorporates everything on a single printed circuit board, including the video display generator. The output of this generator is fed through the black cable coming from the back of the SOL and across the back of the television. This is part of a section of shielded coaxial cable supplied with the SOL-20 kit. It is connected to a small aluminum box containing a Pixe-Verter, a battery pack of 4 AA cells, and an on/off switch (hidden). The radio frequency output from the Pixe-Verter is fed through a twisted pair of solid conductor wires to the small black connector on the back of the television set. This connector was supplied with the set and is used for connecting an external antenna. The upper binding posts are for VHF and the lower are for UHF, the switch in the center is for a local/distant setting (it is set for local).

A Message to our Subscribers

From time to time we make the BYTE subscriber list available to other companies who wish to send our subscribers promotional material about their products. We take great care to screen these companies, choosing only those who are reputable, and whose products, services, or information we feel would be of interest to you. Direct mail is an efficient medium for presenting the latest personal computer goods and services to our subscribers.

Many BYTE subscribers appreciate this controlled use of our mailing list, and look forward to finding information of interest to them in the mail. Used are our subscribers' names and addresses only (no other information we may have is ever given).

While we believe the distribution of this information is of benefit to our subscribers, we firmly respect the wishes of any subscriber who does not want to receive such promotional literature. Should you wish to restrict the use of your name, simply send your request to BYTE Publications Inc, Attn: Circulation Department, 70 Main St, Peterborough NH 03458. Thank you.

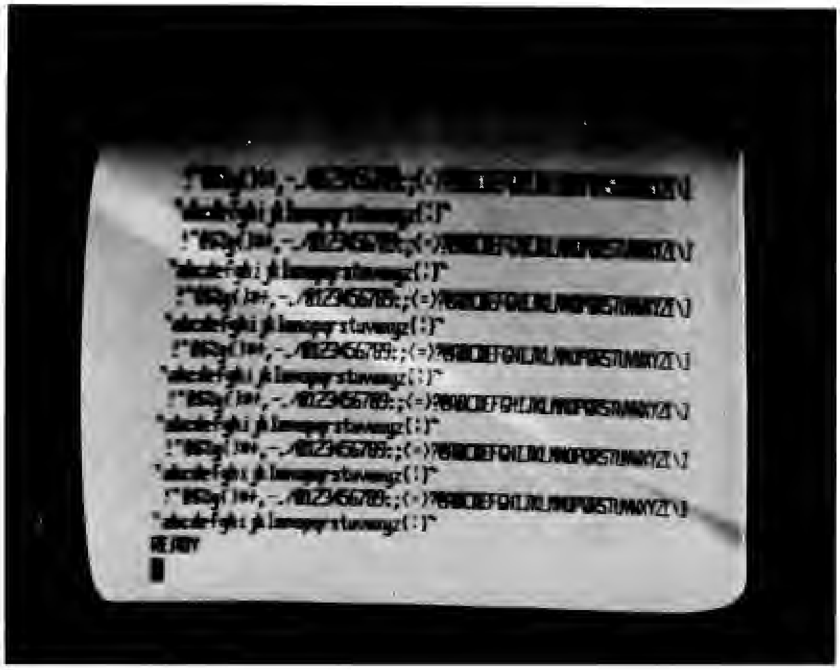


Photo 3: *Selected portion of the available character set.*

The television itself is a Sony Model 9-51UW with a 9 inch (13.5 cm) diagonal screen. It is over 11 years old and well-used.

About the Photos

All photos were taken with a Leica M3 with a 50 mm F/2 dual range Summicron lens, using Tri-X ASA 400 35 mm film. The delayed shutter release was used to dampen vibration effects. Photo 1 was taken with existing light from a window on a sunny day at 1/60 second at f/5.6. Photos 2 and 3 were taken at a distance of 21 inches (53 cm) at 1/30 second at f/4. The bright diagonal bands seen on the screen are due to the discrepancy between the shutter speed, the scan rate of the television, and the focal plane shutter of the camera used to take the photos. The darkening at the top of the screen, the heightening of the characters at the top, and the slight slanting of the characters is due to the poor vertical and horizontal linearity of the set. The slight fuzziness of the display at the left of the pictures is due to depth of field restrictions. Overall, photos 2 and 3 accurately represent what is seen by the human eye and brain (eye persistency eliminates the bright diagonal bands).

Photo 2 is a common display of a portion of a program listing using all upper case characters. Photo 3 is a generated display of a selected portion of the available character set. Control characters were eliminated, since they would cause unwanted display functions like carriage returns and screen clearing. Photos 2 and 3 each show one or more lines with 64 characters each.

Photo 3 is most indicative of the limitations of the radio frequency entry method. The lower case characters *m* and *w* show a definite merging of the dot pattern. The upper case versions also show this effect to a lesser degree. In normal use, however, the human eye and brain manage to fill in gaps in definition.

I hope this material will show that the radio frequency entry method is capable of producing a very satisfactory video display of 64 characters per line.■

Technical Forum is a feature intended as an interactive dialog on the technology of personal computing. The subject matter is open-ended, and the intent is to foster discussion and communication among readers of BYTE. We ask that all correspondents supply their full names and addresses to be printed with their commentaries.

Continued from page 6

space potential of the current 8 bit microprocessors has effectively become saturated.

This saturation of memory address space in the 8 bit 40 pin package microprocessors with a mere eight parts leads to the next new high in semiconductor technology's current innovations: the testing and subsequent approach to volume production of three excellent large scale microprocessors which provide 23 and 24 bit address spaces capable of reaching 8 or 16 million bytes of memory (or peripheral hardware.) I refer of course to the new crop of 16 bit traditional microprocessors introduced by Intel, Zilog and Motorola. Perhaps the first such part was the Intel 8086 announced last spring and most likely in production by the time this is written. (From one contact I heard mention of an even newer 8087, but have not seen any written information on such a part to date.) The second part, announced shortly after the 8086 last spring, is the Zilog Z-8000. But what appeals to my mind, after hearing engineering introductory talks on all three of these new products by representatives of the companies, is the Motorola 68000. It is my own personal favorite, providing a 24 bit byte address space and a relatively simple system design concept without elaborate memory paging and address bus multiplexing requirements. It is the kind of 16 bit microprocessor I like, namely one with a separate 24 bit byte address bus, a 16 bit bidirectional data bus and simple power supply requirements. If I were to build a new system of the homebrew variety at the present time, it is the one I would most likely use. For the moment then, the three processors from Intel, Zilog and Motorola are the best possibilities for overcoming the address limitation problems which become very real as the 64 K dynamic memory parts come to market.

The third major development of the current crop of large scale integration technology is that of new video display controller chips. These parts are actually in production at the present time, and are, no doubt, quietly buried in the designs of many of the personal computing products which have come to the market for the first time in late 1978 and those which will arrive in early 1979. We've already received a number of articles on this kind of device, articles which readers will see in an upcoming special issue on the theme of video graphic interfaces.

The final and most exciting development of recent months was relayed to me by Ken Bowles of the Pascal project at the University of California at San Diego (UCSD) in a phone conversation this past September 20. This is the development of a microcomputer chip set which *directly executes* the UCSD Pascal compiler's p-code intermediate language. The firm responsible for this innovation is Western Digital, 3128 Red Hill Av, POB 2180, Newport Beach CA 92663. In phone conversation with Dr Larry Lotito of Western Digital I found out some more details about the processor, which he and Ken jokingly call a "sand casting" of the UCSD p-code interpreter. This first high level language machine in microprocessor form will come to market in several forms in January of 1979.

As readers familiar with the development of minicomputer technology into microcomputer form will recall, Western Digital was the semiconductor manufacturer which designed and first supplied the chip sets for the Digital Equipment Corp (DEC) LSI-11 product several years ago. These chip sets consist of a microprogram controller and a set of read only memory programs which emulate the desired computer's architecture. After the first LSI-11 parts had been created and marketed, DEC began its own in house semiconductor fabrication efforts and Western Digital turned out to have less of a market for its microprocessors than might have been expected.

In the past two years or so, several variations of the basic 16 bit architecture of this chip set have been offered on a custom basis, and at least one such variation has appeared in the form of an advanced S-100 bus computer (MCP-1600). Of course, Western Digital has continued to supply standard parts for the digital systems markets, such as floppy disk controller chips, and serial communications interfaces, among others. With the experience of producing more than one read only memory microcode definition for the MCP-1600 microprocessor system design, it was not hard for the firm to write the microcode needed to emulate a new design, a "P-engine" that executes the intermediate language codes produced by the Pascal compiler developed by UCSD. Western Digital calls the resulting chip set the "Pascal Micro Engine" and considers this name their proprietary trademark. According to Larry, this product will be widely available in several forms in January of 1979. What is significant is that the software development system for this chip set is the UCSD Pascal system without any modification: a combination of Pascal compiler, editor and disk operating system written in Pascal. The assembler for this machine is the UCSD Pascal compiler, and its "assembly language" is Pascal.

One of the most interesting forms in which the Western Digital Pascal Micro Engine will be made available is as an assembled black box computer which contains the following hardware: the Pascal Micro Engine processor; 64 K bytes of programmable memory; two RS-232 serial terminal ports capable of operations at up to 19,200 bps; two parallel ports for support of a printer; a floppy disk controller with direct memory access which is capable of interfacing four floppy disk drives of 8 or 5 inch size, single or double density; and a power supply for the processor. (Users will thus have to supply a terminal and the actual floppy disk drives with their own power supplies.) Given the special introductory price of $1995 for the first 500 systems produced, and the $2995 price after the introductory period, this package is truly amazing. It is intended as a development system for the MCP-1600 Pascal Micro Engine chip set, even though it will certainly be available through the more progressive local dealers to the personal computer trade.

Western Digital, however, considers itself mainly a semiconductor manufacturer, so one of the reasons for the relatively low price on the development system's processor kernel is to promote sale of the chip sets for use in new designs. We can expect to see more than one personal computer manufacturer taking advantage of this development, for the characteristics of the directly executed p-code method allow programs to run from six to eight times faster than would be possible using the LSI-11 versions of the software.

In the LSI-11 version of the UCSD system (or any other conventional processor's version) there are two levels of emulation. At the first level is the hardware needed to execute the instruction set of the particular microprocessor, be it 8080, LSI-11, 6800, 6502 or any other instruction set. The second level comes in when the particular microprocessor runs an interpreter which emulates the P-machine. With the Western Digital innovation, the P-machine is directly executed by the hardware which is seen by the system designer. This direct execution is the reason for the improvement relative to the LSI-11 which uses very similar hardware. Designers who are interested in creating dedicated microprocessor systems that use the most advanced and reliable software development techniques will find this chip set a natural one to use. Designers of personal computing products will also find it useful, for the extremely powerful UCSD Pascal software system fits naturally into the machine.

This announcement of a high level language machine for Pascal is perhaps the high point of the current crop of wonders which include the 64 K memories, large scale microprocessors and video controllers. Some people have disputed the relevance of high level languages like Pascal, on the ground that they demand expensive systems, but the arrival of the relatively inexpensive Western Digital machine next month is perhaps the last word on that argument for now. The nature of the new levels of sophistication in the larger microprocessor chips such as the 8086, Z-8000 and 68000 complement the new heights of memory density in the 64 K chips and further indicate both the need for and practicality of high level languages like Pascal in future personal computers.■

Technical Forum

Resonating Modulators

Allen Watson III
430 Lakeview Way
Redwood City CA 94062

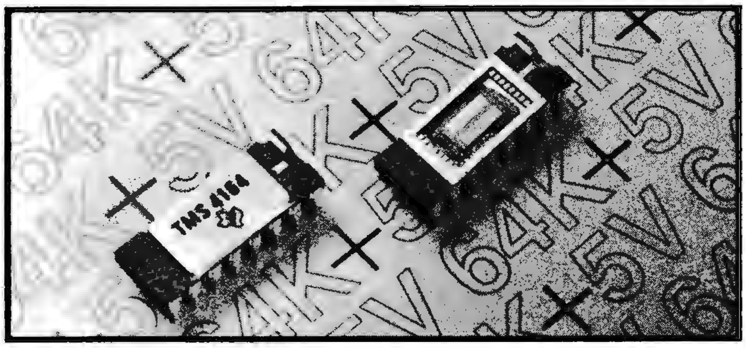
As a recent Apple II purchaser, I enjoyed your review of the Apple II (March 1978 BYTE, page 18). I was especially interested to see that you encountered most of the same problems I did, such as the interference with the color receiver. I too am using the M & R Enterprises modulator that installs inside the Apple II along with a low priced GE portable, and the interference with my first setup was pretty bad.

Something about the length of the lead to the TV set got me to thinking. I remember making dipole antennas, and somehow 4 feet seemed like a familiar dimension. Channel 3's picture carrier frequency is about 61 MHz, for a wavelength of about 5 meters. The cable from the modulator to the color set is about 4 feet (1.2 meters) long or almost exactly a quarter wave. That makes it a very good antenna for any harmonics (60 thru 65 MHz) of the Apple clock, character generator, etc. The cable is looped through a large ferrite toroid, which helps quite a bit, but a simple modification makes things even better. All you have to do is add an 18 inch extension cable, thus mistuning the channel 3 antenna, and 90 percent of the interference will disappear.

*We just got around the problem of radio frequency interference with our Apple II by use of the M & R Enterprises UHF modulator recently acquired. Without even putting a single toroidal balun core on the coaxial cable, the same Panasonic color television runs without any interference. . . .*CH■

What's New?

Texas Instruments Introduces 5 V 64 K Byte Programmable Memory



Texas Instruments has introduced the TMS 4164, a single 5 V 64 K byte dynamic programmable memory, organized as 64 K byte by 1. It comes in a 16 pin dual-in-line package, and allows upward compatibility with the 16 K byte dynamic programmable memory.

The TMS 4164 single 5 V power supply design is TTL compatible, offers lower power dissipation, and is more immune to system noise. Compact layout, and an optimized design and process combination for 5 V only operation result in improved performance.

Access times range from 100 to 150 ns maximum, with minimum cycle times of 200 to 250 ns. Power dissipation is 200 mW maximum or 3 μW maximum per bit. Comparing the 462 mW power dissipation of the 16 K programmable memory at 375 ns cycle time, total maximum power dissipation of the new memory is a reduction of 60 percent, with improved cycle times, while bit density is quadrupled. As a result of the lower power dissipation, the TMS 4164 features a 256 cycle refresh with a 4 ms maximum refresh period.

Due to TMS 4164 refresh compatibility with the 16 K byte programmable memory, the basic refresh controller timing does not require major changes. The only provision required is for an 8 bit refresh counter and multiplexer when upgrading to 64 K byte from a 16 K byte system. Also contributing to higher system operating efficiency is a 1.3 to 1.6 percent refresh overhead time, compared to 2.4 percent on the 16 K byte programmable memory.

The TMS 4164 is priced at $125. For further information write to Texas Instruments Inc, Inquiry Answering Service, POB 1443, M/S 669, Houston TX 77001.■

Circle 593 on inquiry card.

A New Color Graphics Computer from Chromatics

Chromatics Inc has introduced the CG series line of full 8 color graphic and alphanumeric readout computers. The line consists of 13, 15 and 19 inch models featuring noninterlaced screen refresh, high resolution shadow tubes, and 512 by 512 or 512 by 256 individually addressable and color selective dots. Each model employs a Z-80 processor with full memory and input and output (IO) structure. The 13 inch model starts at $8995.

A bulletin describing the system may be obtained from Chromatics Inc, 3923 Oakcliff Industrial Ct, Atlanta GA 30340.■

Circle 594 on inquiry card.

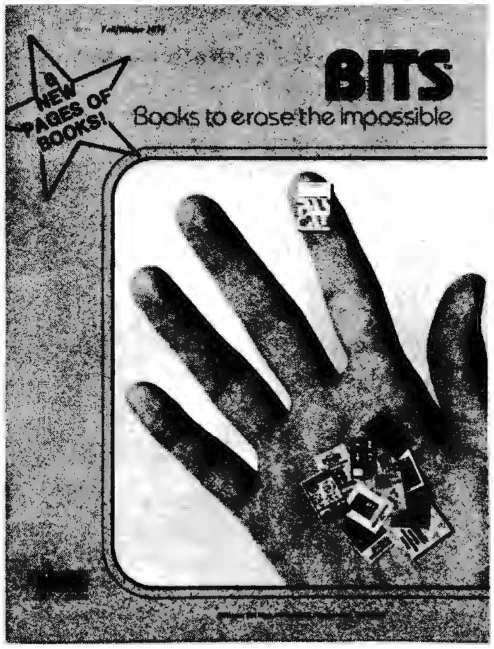
New Keyboard for 64 and 80 Character Display Video Boards

This new MKB-2 keyboard is designed for use with the 64 and 80 character display video boards. Standard features on the MKB-2 include: a numeric key pad, upper and lower case, cursor control keys, 2 key rollover, and automatic repeat on all keys. The MKB-2 is assembled in a heavy duty steel case with parallel interface, strobe or pulse and on board regulation (5 V, 12 V), and comes complete with standard DB25S connector and black double injection molded keys.

The price of the MKB-2 is $149. For further information, write to MicroAge, 1425 W 12th Pl #101, Tempe AZ 85281.■

Circle 595 on inquiry card.

Expanded Book Catalog from BITS



A new expanded BITS catalog is now available, featuring books on microcomputing and other related subjects. There are new books on business and calculators, Pascal, artificial intelligence, robotics, programming, hardware, games and much more.

BITS has a complete selection of professionally reviewed microcomputer books (over 150 titles), including a number of self-published works. Their inventory has been expanded to include a greater number of posters and specialty items of interest to the computer enthusiast. Request your free catalog from BITS Inc, Dept 3, POB 428, Peterborough NH 03458.■

Circle 564 on inquiry card.

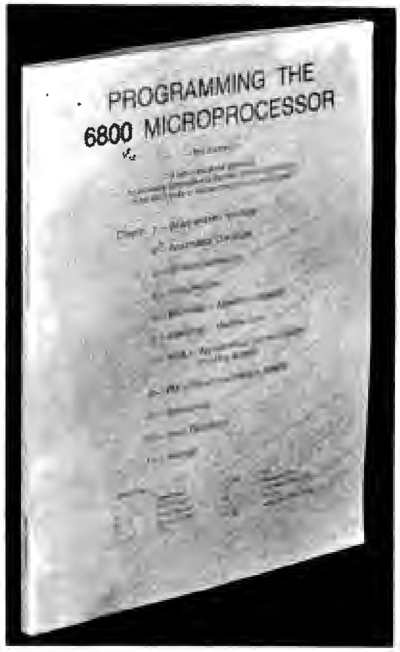
Software Handbook of Statistical Techniques

This software handbook, entitled *SIMPLE*, comprises over 100 conversational computer programs written in BASIC for teaching statistics and experimental design to students from all engineering, science, business and social science disciplines. Course organization around the modular structure is considered; and the manual contains a description of each program, the program input and format requirements, a sample problem, a copy of the terminal user dialogue and an explanation of what the results mean in practice and the BASIC program itself.

The price for this manual is $10.95. For further information contact Sterling Swift Publishing Company, POB 188, Manchaca TX 78652.■

Circle 565 on inquiry card.

6800 Programming Workbook



This self-instruction workbook is a guide to the fundamentals of assembly language and machine code programming of the 6800 microprocessor and its peripheral devices. Considerable coverage is given to programming of input and output devices. The asynchronous communications interface adapter and peripheral interface adapter, each with its various modes of operations, are explored in detail in both noninterrupt and interrupt modes. Program design and documentation are emphasized along with programming hints and aids. *Programming the 6800 Microprocessor* by R W Southern is available from Southcroft Publications, POB 11703, Station H, Ottawa CANADA K2H 7T8, for $6.75. Submit US funds for US delivery; Canadian funds for Canadian delivery.■

Circle 566 on inquiry card.

New Publication Devoted to TRS-80 User

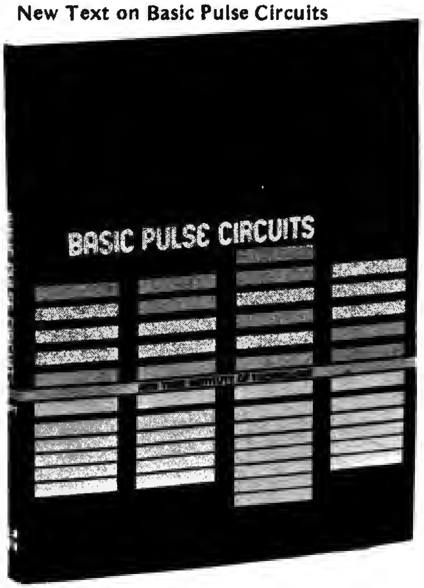
Dump Publications has announced the release of a software publication for users of the Radio Shack TRS-80 microcomputer system. *Dump* is a monthly periodical incorporating news, information, and running software ready to load from a 33 1/3 revolution per minute disk record. The *Dump* disk can be loaded into the TRS-80 system with the use of an ordinary phonograph.

Each issue contains a wide variety of programs from finance and education to games and machine language. Programs are provided with complete documentation and line editing information for Level I and II BASICs.

A 1 year subscription costs $20. For more information, contact Dump Publications, POB 2454, Jacksonville FL 32203.■

Circle 567 on inquiry card.

New Text on Basic Pulse Circuits

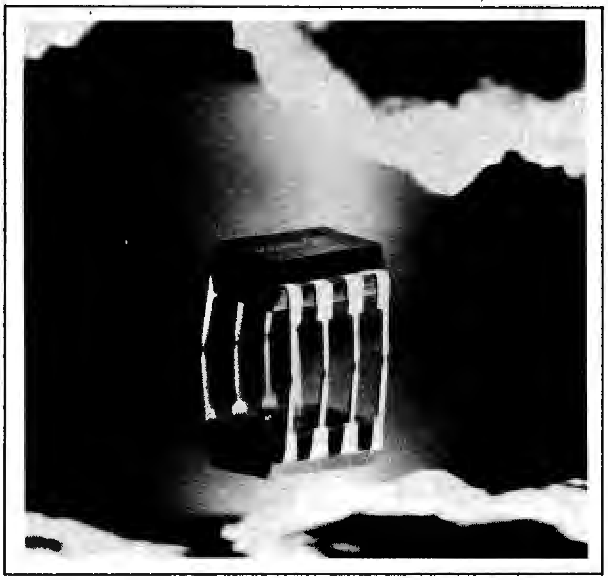


The basic building blocks of modern computers, radar, television and pulse communication circuits are presented in this programmed text entitled *A Programmed Course in Basic Pulse Circuits*, by the New York Institute of Technology.

This 293 page programmed learning text is organized in a logical sequence of interrelated steps. Discussions on switching devices such as unijunction transistors and silicon controlled rectifiers are included. All devices are solid state, and some material on integrated circuits is presented. Each chapter begins with a set of objectives and concludes with a set of criteria tests to measure progress.

The price of this text is $9.95 and it can be obtained from McGraw-Hill Book Company, 1221 Av of the Americas, New York NY 10020.■

Circle 568 on inquiry card.



THE BEST FOR BOTH WORLDS

Factory prime electronic equipment for both the discerning hobbyist and the professional

The New Hobby World Catalog

Your source for factory prime, professional quality equipment. Computers, add-on boards, IC's, sockets, resistors, supplies, tools, test equipment, books, and more. Shop your buy list at Hobby World. You'll find what you want, and at a solid savings.

This month's specials.

16K MEMORY ADD-ON FOR APPLE OR TRS-80
Hobby World price is only **$99.00** (specify when ordering)

NEW FROM SSM. THE CB-1 8080 CPU BOARD AT A HOBBY WORLD SUPER SPECIAL. (See SSM ad for specs) It's Loaded. It's Blue. IT'S ONLY **$119.95**

THE NEW ELENCO 3½ DIGIT SOLID STATE MULTIMETER
The ultimate in performance: measures resistance to .01 ohms, voltage to 100 micro-volts, current to one micro-amp. Assembled and tested, with 2-Year warranty. Lists at **$99.95.** Hobby World: **$74.95**

WAHL ISO-TIP CORDLESS SOLDERING IRON
Includes ni-cad batteries and wall plug transformer. Lists at **$19.95,** but Hobby World has it for **$14.95!**

S-100 WIRE WRAP BOARD
Over 3600 holes, for super versatility. Provisions for 4 regulators. Double-sided, plate-through. All S-100 pins labelled for number and signal. The PT-1. Hobby World Price: Only **$30.00**

Hobby World
19355 Business Center Drive #6
Northridge, CA 91324

TELEPHONE ORDERS
Inside Calif: 213 886-9200
Outside Calif: 800 423-5387

Hobby World Carries TAB Books! Affordable books for the hobbyist. Hobby World helps out with a 10% discount. (Order by number, and don't forget the 10% discount)

		LIST PRICE
Computer Circuits and How They Work	#538	**$5.95**
Beginner's Guide to Computer Logic	#548	**$5.95**
Computer Technician's Handbook	#554	**$9.95**
Build Your Own Working Robot	#841	**$5.95**
Programming Microprocessors	#985	**$6.95**
Beginner's Guide to Computer Programming	#574	**$9.95**
Digital Electronics: Principles and Practice	#585	**$5.95**
Microprocessor/Micro-programming Handbook	#785	**$6.95**
Computer Programming Handbook	#752	**$9.95**

Order by mail. Just write. Or order by phone. Toll free outside California.

SEND ME A CATALOG!

Name

Address

City State

Satisfaction 100% Guaranteed

BANKAMERICARD VISA

master charge

SEND ME THE FOLLOWING SPECIALS!

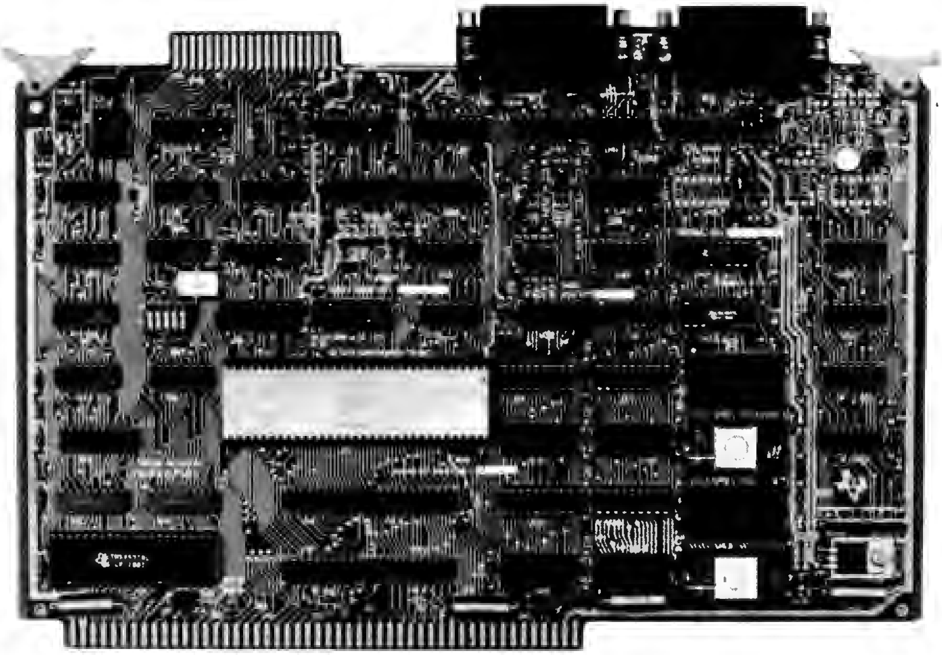
Item	Quantity	Price
	Postage/Handling	$1.50

California Residents add 6% Sales Tax
Charge My MC VISA #

Signature

Expiration

16 Bit Microcomputer Module



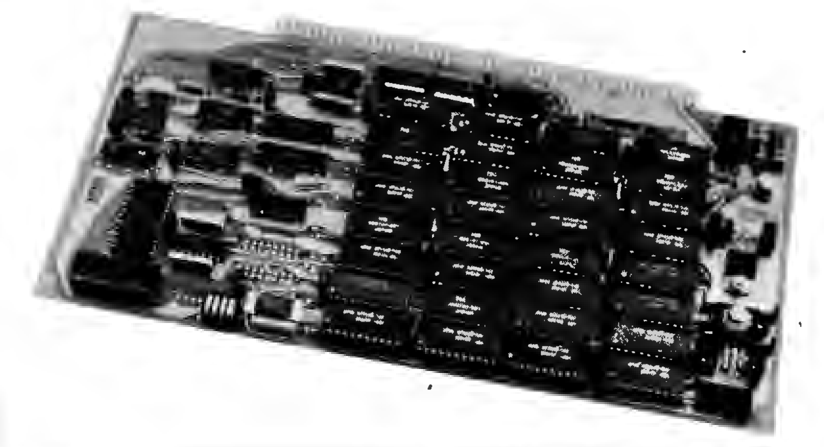
The TM 990/100M series offers up to 4 K words by 16 bits of erasable read only memory and up to 2 K words by 16 bits of static programmable memory on board. The board also contains two serial input and output (IO) ports; one is intended for remote usage with a terminal or modem, the other for local usage with Texas Instruments' 301 Microterminal, an EIA terminal or a Teletype.

The TM 990/101M series offers three programmable interval timers, up to 17 interrupts and 16 lines of programmable parallel IO. The TM 990 series is supported by Texas Instruments' AMPL prototyping system. A user's manual detailing the hardware and software of the TM 990 board will be supplied with each unit. A wide line of accessories and peripherals also available.

Pricing on the fully assembled and tested TM 990/101M-1 including 1 K word by 16 bit erasable read only memory and 1 K word by 16 bit static programmable memory is $625 in single quantities. Inquiries should be forwarded to Texas Instruments Inc, Inquiry Fulfillment, POB 1443, M/S 653 (Attn: TM990), Houston TX 77001.■

Circle 555 on inquiry card.

S-100 Compatible Single Card Plug-In NMOS Memory



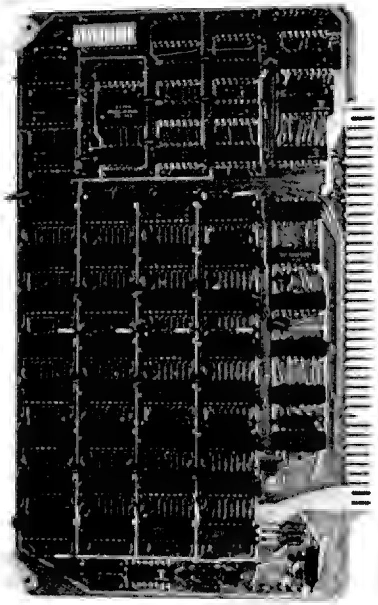
This 16 K word by 8 bit programmable NMOS memory system is S-100 and card size compatible. The EMM Model 1104 is a single card plug-in assembly which is fully burned-in and tested. The Model 1104 uses EMM 4 K byte static programmable memories, and no refresh circuitry is required.

The system consists of the memory array with support electronics including address and data buffering, timing and control, and voltage regulation. The memory array is divided into four 4 K by 8 bit memory blocks, and each block can be assigned to a 4 K byte address block within a 0 to 64 K byte range. Operating modes are read, write and deposit. The deposit mode is a phase memory cycle consisting of a write followed by a read.

For complete information contact Electronic Memories and Magnetics Corp, 12621 Chadron Av, Hawthorne CA 90250.■

Circle 557 on inquiry card.

High Speed EXORcisor Memory



Designed specifically for operation with Motorola EXORcisor and MEC 6800 evaluation module is Chrislin Industries' new CI6800, 16 K by 8 bit semiconductor memory system. The new memory allows expansion to 32, 48 and 64 K bytes by interchanging the 4027 4 K by 1 bit dynamic memory with the 16 K equivalent. No further modification is required. The CI6800 memory board plugs directly into existing EXORcisor connectors.

It allows maximum processor throughput with the use of hidden refresh control logic on board. Data access time is 300 ns and cycle time is 750 ns.

On board memory select is available in 4 K byte increments up to 64 K words of memory. A write disable switch on board makes the programmable memory a read only memory to the outside world.

Complete board power consumption is under 5 W. The board size is 5.75 by 9.75 inches (14.61 by 24.77 cm). For more information, contact Chrislin Industries Inc, 31312 Via Colinas #102, Westlake Village CA 91361. ■

Circle 556 on inquiry card.

PLACE ORDERS TOLL FREE:

800/421-5809 Continental U.S.
800/262-1710 inside California

JADE Computer Products

EPROM BOARD KITS

RAM'N'ROM (16 K any EPROM)	$117.00
MR-8 (1K RAM, uses 2708)	$99.50
MR-16 (1K RAM, uses 2716)	$99.50
EPM-1 (uses up to 4K of 1702)	$59.95
JG8/16 (uses 2708 or 2716)	$69.95

MICROPROCESSORS

F8	$16.95
Z80 (2MHz)	$20.00
Z80A (4MHz)	$25.00
CDP1802CD	$17.95
2650	$19.95
AM2901	$20.00
6502	$11.95
6800	$16.95
6802	$25.00
8008-1	$12.95
8035	$20.00
8035-8	$21.00
8080A	$10.00
8085	$23.00
TMS9900TL	$49.95

8080A SUPPORT DEVICES

8212	$2.90
8214	$4.65
8216	$2.75
8224 (2MHz)	$4.30
8224-4 (4MHz)	$9.95
8226	$2.75
8228	$6.40
8238	$6.40
8243	$8.00
8251	$7.50
8253	$20.00
8255	$8.50
8257	$20.00
8259	$20.00
8275	$75.00
8279	$18.50

USRT

S2350	$10.95

UARTS

AY5-1013A	$5.25
AY5-1014A	$8.25
TR1602B	$5.25
TMS6011	$5.95
IM6402	$9.00
IM6403	$9.00

BAUD RATE GENERATORS

MC14411	$10.00
14411 Crystal	$4.95

6800 PRODUCT

6810P	$4.00
6820P	$6.60
6821P	$6.60
6828P	$11.25
6834P	$16.95
6850P	$8.65
6852P	$11.00
6860P	$9.25
6862P	$12.00
6871P	$28.75
6875P	$8.75
6880P	$2.50
68810P	$6.00

KIM SUPPORT DEVICES

6102	$8.00
6502	$11.95
6520	$10.00
6522	$9.25
6530	$15.95
6530-002	$15.95
6530-003	$15.95
6530-004	$15.95
6530-005	$15.95
6532	$17.95

CHARACTER GENERATORS

2513 Upper (-12+5)	$6.75
2513 Lower (-12+5)	$6.75
2513 Upper (5 volt)	$9.75
2513 Lower (5 volt)	$10.95
MCM6571 up scan	$10.95
MCM6571A down scn	$10.95

PROMS

1702A	$8.00
2708	$9.95
2716 (5+12) TI	$25.00
2716 (5v) INTEL	$60.00
2758 (5v)	$23.40

DYNAMIC RAMS

416D/4116 (250ns)	$12.50
2104/4096	$4.00
2107B-4	$3.95
TMS4027/4096 (300ns)	$4.00
MM5270	$4.50
MM5280	$3.60

STATIC RAMS

	1–15	16-100
21L02 (450ns)	$1.50	$1.20
21L02 (250ns)	$1.75	$1.50
2101-1	$2.95	$2.60
2111-1	$3.25	$3.00
2112-1	$2.95	$2.65

FLOPPY DISK CONTROLLERS

1771B01	$39.95
1781	$69.95

KEYBOARD CHIPS

AY5-2376	$13.75
AY5-3600	$13.75
MM5740	$18.00
MM5743	$18.00

EXPANDOR'S BLACK BOX PRINTER

This 64-character ASCII impact printer with 80-column capability is portable and uses standard 8½" paper and regular typewriter ribbon. Base, cover and parallel interface are included. Assembled and complete with manual and documentation.

only $430.00
(90 day manufacturer's warranty)

TRS–80 Interface Cable for Black Box Printer
with mating connectors: $48.00
(must be used with expansion module, +8v/1 amp power supply required).

Power Supply for TRS–80/Black Box Printer $49.00

★ STATIC RAM SPECIALS ★

2114's, low power (1024x4)

	1-15	16-99	100 +
450ns	8.00	6.95	5.50
250ns	9.00	8.00	6.50
TMS4044/MM5257, low power			
450ns	8.00	7.50	6.50
250ns	9.95	8.75	8.00
4200A (4Kx1, 200ns)			
	9.95	8.50	8.00

Look!

DYNAMIC RAM BOARDS EXPANDABLE TO 64K

32K VERSION • KITS

Uses 4115 (8Kx1, 250ns) Dynamic RAM's, can be expanded in 8K increments up to 32K.

8K	$199.00
16K	$255.00
24K	$310.00
32K	$369.00

64K VERSION • KITS

Uses 4116 (16Kx1, 250ns) Dynamic RAM's, can be expanded in 16K increments up to 64K.

16K	$292.00
32K	$432.00
48K	$578.00
64K	$721.00

JADE PARALLEL/SERIAL INTERFACE

S-100 compatible, 2 serial I/O ports, 1 parallel I/O.

Kit JG-P/S	$124.95
Ass. & Tested JG-P/SA	$179.95
Bare Board with manual	$30.00

FLOPPY DISK INTERFACE

JADE FLOPPY DISK (Tarbell board)

Kit	$175.00
Assm. & Tested	$250.00

S.D. Computer Products "Versa Floppy"

Kit	$159.95
Assm. & Tested	$189.95

STATIC RAM BOARDS

JADE 8K

Kits: 450ns	$125.95
250ns	$149.75
Assembled & Tested:	
450ns	$139.75
250ns	$169.75
Bare Board:	$25.00

16K - Uses 2114's (low power)

Assembled & Tested:	
RAM 16 (250ns)	$375.00
RAM 16B (450ns)	$325.00
MEM-2 Kit:	
250ns	$285.00

16K with memory management

Assembled & Tested:	
RAM 65 (250ns)	$390.00
RAM65B (450ns)	$350.00

32K Static

Assembled & Tested:	
250ns	$795.00
450ns	$725.00
250ns Kit	$575.00

TRS-80 apple II MEMORY EXPANSION KITS

4116's (16Kx1, 250ns)

8 for $85.00

TRS–80 KIT
includes connectors and instructions $98.00

JADE 8080A

with full documentation

Kit	$100.00
Assm. & Tested	$149.95
Bare Board	$30.00

JADE Z80

with provisions for ONBOARD 2708 and POWER ON JUMP

2 MHz

Kit	$135.00
Assm. & Tested	$185.00

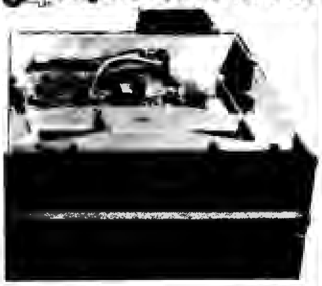
4 MHz

Kit	$149.95
Assm. & Tested	$199.95
Bare Board	$35.00

MD-690a CPU BOARD

S-100 Compatible 6800 MPU 1K x 8 RAM, PROM expandable to 10K. $199.95

Shugart DISK DRIVES



SA 400 $295.00
Single-density 5¼", 35 track drive.

SA801R $495.00
Single-sided 8" floppy disc drive.

DM 2700-S $750.00
Includes SA801R disk drive, 10" x 10" x 16" cabinet, power supply, data cable, fan, AC line filter.

S–100 MOTHER BOARDS

JADE 6-SLOT

Kit	$41.95
Assembled & Tested	$56.95
Bare Board	$24.95

9–SLOT "LITTLE MOTHER"

Kit	$85.00
Assembled & Tested	$99.00
Bare Board	$35.00

13–SLOT "QUIET MOTHER"

Kit	$95.00
Assm. & Tested	$110.00
Bare Board	$40.00

22–SLOT "STREAKER"

Assm. & Tested	$149.00

JADE VIDEO INTERFACE

S-100 Compatible Serial Interface with Sockets Included.

Kit	$117.95
Assm. & Tested	$159.95
Bare Board with manual	$35.00

ALL PRICES SUBJECT TO CHANGE WITHOUT NOTICE

JADE Computer Products
4901 W. ROSECRANS AVENUE
Department "B"
HAWTHORNE, CALIF. 90250

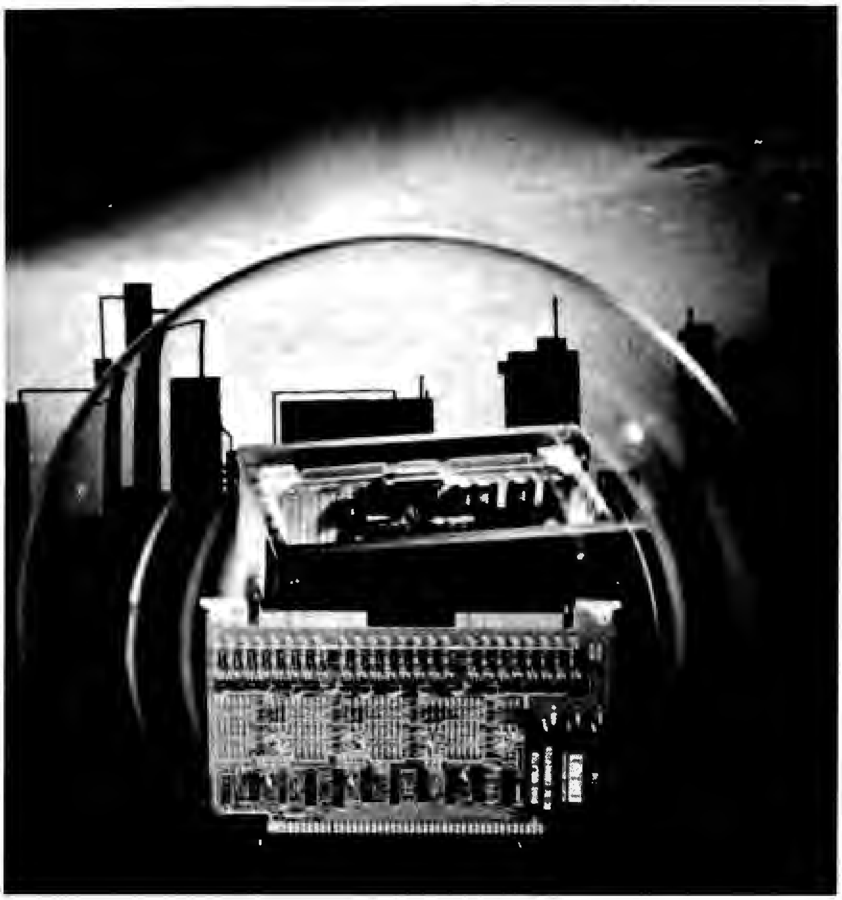
THE PIGGY IS COMING!

Cash, checks, money orders, and credit cards accepted. Add freight charge of $2.50 for orders under 10 lbs and $1.00 service charge for orders under $10.00. Add 6% sales tax on all parts delivered in California. Discounts available at OEM quantities.

WRITE FOR OUR FREE CATALOG

Circle 195 on inquiry card.

24 Channel Digital Input System for Motorola Microcomputers



This single board microperipheral accepts 24 digit inputs. MP710, with an on board power supply, operates with dry relay contacts and MP710-NS with voltage input (wet relay contacts). Each group of eight inputs is isolated from other groups and from the computer bus up to 600 VDC. In MP710-NS, isolation between inputs is 300 VDC.

MP710s are electrically and mechanically compatible with Micromodule and EXORciser microcomputers and operate from their +5 VDC supplies. They are programmed as memory locations and with each input using one memory bit, any read command may be used. When the board is read, logic 0 represents an open contact (low voltage); logic 1, a closed contact (high voltage). Each read command inputs the status of eight channels. Address bits A0 and A1 select the set of inputs to be read. The remainder of the address lines are used to select the board itself. The address block occupied by each board is selectable and can be located anywhere in memory.

The price of the MP710 is $355 in quantities of one to nine. For further information contact Burr-Brown, International Airport Industrial Park, Tucson AZ 85734.■

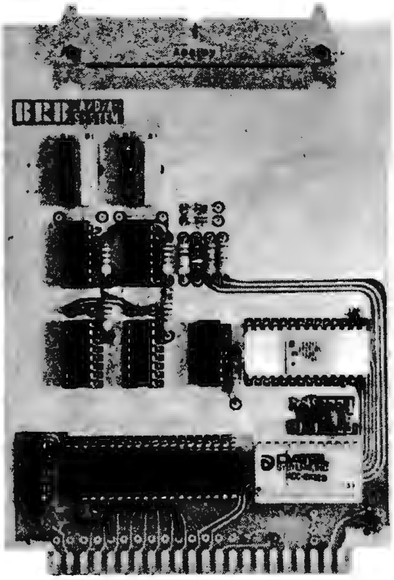
Circle 551 on inquiry card.

Process Control Output Module

The PCO-1A process control output module provides two complete 4 to 20 mA or 10 to 50 mA process control circuits on one Wyle microcomputer system output module. The 4 to 20 or 10 to 50 ranges are independently selectable for each circuit, and both outputs are short circuit protected. The PCO-1A is priced at $345 per module (2 output circuits). Contact Wyle Laboratories/Computer Products, 3200 Magruder Blvd, Hampton VA 23666.■

Circle 552 on inquiry card.

Microprocessor Analog Interface Module



The Wince Analog Interface Module enables laboratory and control engineers to interface thermocouples and other transducers to a microprocessor and interface the microprocessor to motors, servos, etc. Options include a 16 channel multiplexer, an 8, 10 or 12 bit analog to digital converter and one or two 8 bit digital to analog converters. The base price is $99. Write to Wintek Corp, 902 N 9th St, Lafayette IN 47904 for further information.■

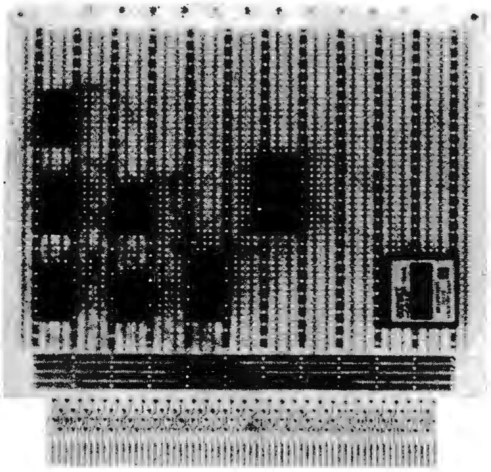
Circle 553 on inquiry card.

Computer Video to UHF RF Interface Modulator

The Micro-Verter is designed to interface microcomputers to color or monochrome television receivers as an alternative to the video monitor. The Micro-Verter operates in the UHF channels above channel 14, beyond the normal range of switching harmonics, and is designed to interface directly with the Apple II as well as with most other microcomputers. It comes complete with video cable and radio frequency (RF) output stub coupler and requires no direct connection to antenna terminals except in special cases. The radio frequency signal is coupled directly into the UHF tuner input via a 1 cm stub coupler on the back of the modulator. The approximate size of the unit is 2 by 3.5 by 4.5 inches (5.5 by 8.5 by 11.5 cm) and it is priced at $35. For more information contact ATV Research, 13th and Broadway, Dakota City NE 68731.■

Circle 554 on inquiry card.

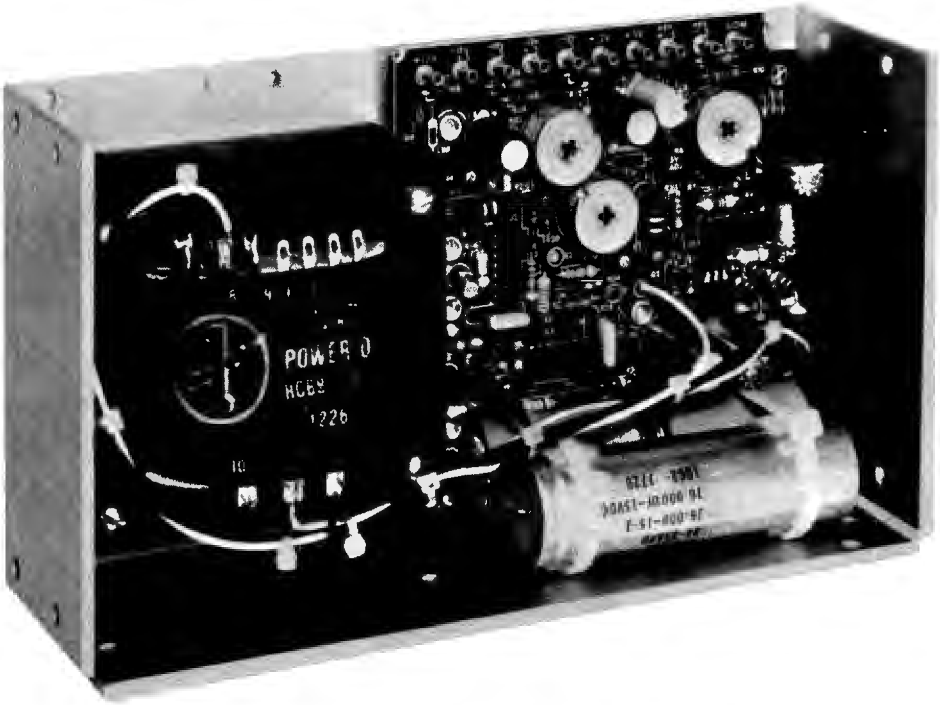
Z-80 Arithmetic Processing Unit



Fully compatible with the Zilog Z-80 MCB, this high speed arithmetic processing unit board (HAPUB) provides the hardware necessary to accomplish arithmetic, trigonometric, inverse trigonometric, logarithmic, exponential and square root functions. HAPUB simplifies software and allows the Z-80 to perform other operations while accomplishing these functions. Also featured are fixed point integer single and double precision (16 and 32 bit), and floating point single precision (32 bit) operation with bidirectional conversion capability. The board is compatible with the Zilog Z-80 card cage and 8 bit bidirectional data bus and costs $749. Contact Signal Laboratories Inc, 202 N State College Blvd, Orange CA 92668.■

Circle 525 on inquiry card.

Triple Output DC Power Supply



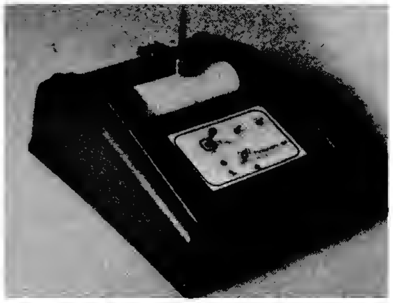
Power-One has announced an addition to their Hi-Vol series triple output DC power supply line. The new model, designated HCAA-60W, is built in the industry standard package size for a 60 W triple output open frame power supply. This model outputs 5 V at 6 A with adequate overvoltage protection, +12 to 15 V at 1.0 A, and −12 to 15 V at 1.0 A. The −12 to 15 V output may be changed to −5 V at 0.4 A by jumpering two printed circuit board terminals. Targeted for use in systems requiring multiple DC voltages, the HCAA-60W will power combinations of most semiconductor devices including TTL, PMOS, NMOS, CMOS and linear devices. Total isolation between the 5 V, ±12 V and ±15 V outputs allows the user to arrange polarities to suit specific applications.

Standard features include 115/230 VAC ±10% AC input capabilities, ±0.05% line and load regulation, and full protection against short circuit and overload. Maximum output ripple is 3 mV peak to peak.

Each unit is tested and burned in and carries a 2 year warranty. The size is 9.0 by 4.87 by 3.2 inches (22.86 by 12.37 by 8.13 cm) and it weighs 7.5 pounds (3.36 kg). The price is $84.95 from Power-One Inc, Power-One Dr, Camarillo CA 93010.■

Circle 527 on inquiry card

Hard Copy Graphics Terminal



This plotter system, called Panographic-84, has a resolution of 100 steps per inch in the X and Y directions and a cumulative error of less than .020 inches (.05 cm) in 10 inches (25.4 cm) of travel. The drives are stepping motor operated for zero drift and no adjustments. Interfacing with a computer is via eight wires from the plotter to a parallel port. When driving the plotter from BASIC language programs, complete handshaking is not required since the plotter response is considerably faster than the speed at which BASIC can drive it. If the user wishes to drive the plotter from a machine language routine, full handshake capability is available. The polarity of handshake signals is switch selectable.

Options available at present consist of a computer operated pen lifter and a vacuum formed plotter cover. Software provided with the system is written in BASIC and listings of these short routines are provided.

The price for the plotter kit without pen lifter and cover is $995. The pen lifter kit sells for $85, as does the molded plotter cover. A factory assembled plotter with pen lifter and cover sells for $1400. For more information, write to Pan Dynamics Inc, 2950 Nebraska Av, Santa Monica CA 90404.■

Circle 526 on inquiry card.

Where Do New Products Items Come From?

The information printed in the new products pages of BYTE is obtained from "new product" or "press release" copy sent by the promoters of new products. We openly solicit releases and photos from manufacturers and suppliers to this marketplace. The information is printed more or less as a first in first out queue, subject to occasional priority modifications. While we would not knowingly print untrue or inaccurate data, or data from unreliable companies, our capacity to evaluate the products and companies appearing in the "What's New?" feature is necessarily limited. We therefore cannot be responsible for product quality or company performance.

California Digital

Post Office Box 3097 B • Torrance, California 90503

CLARE·PENDAR General Instrument Corp.

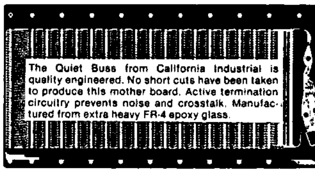
KEYBOARD ASCII ENCODED

63 KEY $64.95

This is a one time purchase of NEW Surplus keyboards, recently aquired from the Telecommunications Division of the Singer Corporation.
The keyboard features 128 ASCII characters in a 63 key format, MOS encoder circuitry "N" key rollover, lighted shift lock, control, escape and repete functions.
Sloped pannel and positive feel switches, makes this professional quality keyboard an excellent buy at only $64.95.
Limited Quantities.

S-100 Mother Board

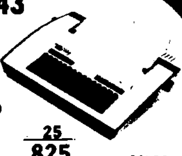
Quiet Buss $29.95 8803-18 18 slot IMSAI



The Quiet Buss from California Industrial is quality engineered. No short cuts have been taken to produce this mother board. Active termination circuitry prevents noise and crosstalk. Manufactured from extra heavy FR-4 epoxy glass.

TELETYPE MODEL 43

Even if we have to give them away, we're going to ship more 43's in 1979 than the aggregate of all our competitors.



Model 43AAA (TTL)

EACH	3	10	25
$925.	875.	850.	825.

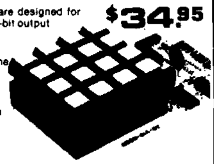
plus shipping

RS-232 Interface "K" Add $75.00

HEXADECIMAL KEYBOARD

$34.95

Maxi-Switch hexadecimal keyboards are designed for microcomputer systems that require 4-bit output in standard hex code.
Each assembly consists of 16 hermetically sealed reed switches and TTL "one shot" debounce circuitry.
Reliable low friction acetal resin plungers are credited for the smooth operation and long life of this premium keyboard.
Requires single + 5 volt supply.



$29.95 BOX of 10 DISKETTES

Verbatim. APPLE/TRS-80 Mini - Soft sector

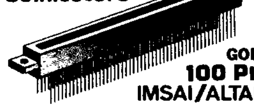
CONNECTORS

your choice DB25P male plug & hood or DB25S female

$3.95

Qty.	fe.	male	hd.
10	3.45	2.45	1.15
25	3.15	2.25	1.05
100	2.85	1.90	.95
500	2.25	1.60	.85
1K	1.97	1.37	.73

Edge Connectors



GOLD 100 PIN IMSAI/ALTAIR

Imsai solder .125x.250	$4.95	3/$10.00
Imsai w/w .125centers	$4.95	3/$13.00
Altair soldertail .140row	$5.95	3/$15.00
SPECIALS		
22/44 Kim eyelet.156"	$1.95	3/$5.00
25/50 solder tab .156"	$1.09	3/$2.00
36/72 wide post w/w.156	$1.95	3/$5.00

$24.88

SPERRY UNIVAC KEYBOARD

The famous Sperry Univac 1710 Hollerith keyboard assembly is now available from California Industrial for only $24.88. The ideal computer input device for accountants and mathematicians. The numeric keys are placed on the lower three rows to resemble a ten key adding machine. This format allows one handed numeric data entry.
Original cost was $385. Used but guaranteed in excellent condition. Complete with documentation.

FROM ATARI

COLOR TELEVISION R.F. MODULATOR

$13.95

Assembled

The Atari R.F. Modulator allows computer data to be displayed directly upon your existing television system. This unit converts the signal from the Apple II and other video sources into television frequencies. Operates from single 5 volt supply. Complete with metal case, mating R.F. connector and 15 feet of coax cable. Schematics and instructions included.

SPECIAL

APPLE II 16K MEMORY

COLOR • GRAPHICS • SOUND

$1024 PLUS SHIPPING

Mfg.Sug. Retail.... $1195

$4.98 10 for $45.

Scotch BRAND Certified Digital CASSETTES

Won't drop a BIT!

$5.50

DISKETTES
8 inch Soft (IBM)
8 inch 32 sector
Mini Soft sec.
Mini 10 sector
Mini 16 sector

CALIFORNIA INDUSTRIAL is an Authorized Dealer of Scotch Brand Data Products

APPLE RS-232

Serial Interface $59.95

Interfaces Apple II to Teletype or other serial printer.

ELECTRONIC SYSTEMS

LED DIGITAL Darkroom Timer • Kit

As featured in the August '78 issue of POPULAR ELECTRONICS.

T'me 1 second to 10 hours

$34.95 Information upon request.

Extender Board

Mullen $34.95

Facilitates design and troubleshooting of all S-100 microsystems. Includes logic probe along with high-low and pulse LED display.
Also available, the Mullin CB-1 controller board $88.

S-100 PROTOTYPE BOARD

$19.98

GP100-Maximum design versatility along with standard address decoding and buffering for S100 systems. Room for 32 uncommitted 16 pin IC's, 5 bus buffer & decoding chips, 1 DIP address select switch, a 5 volt regulator and more.
WW100-Wire wrap breadboard, similar to the GP100. Allows wire wrap of all sizes of sockets in any sizes of sockets in any combination. An extra regulator position for multiple voltage applications.

MEMORY

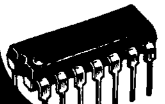
DYNAMIC		1-7	8-32	32+
4115	8Kx1	11.95	11.50	*
4116	16Kx1	13.95	13.00	12.25
(Apple II & TRS80)				
4164	64Kx1	*	*	*

• As you may be aware, publishers require advertisers to submit their ad copy 60 to 90 days prior to "press" date. That much lead time in a volatile market place, such as memory circuits, makes it extremely difficult to project future cost and availability.
To obtain the best pricing on memory we have made volume commitments to our suppliers, which in turn affords us the opportunity to sell these circuits at the most competitive prices. Please contact us if you if you have a demand for volume state of the art memory products.

STATIC	1-31	32-99	100-5C	-999	1K+
21L02 450nS.	1.49	1.19	1.05	.95	.89
21L02 250nS.	1.69	1.49	1.45	*	*
2114 1Kx4 450	6.95	6.50	6.25	6.00	5.75
2114 1Kx4 300	8.95	8.50	8.00	*	*
4044 4Kx1 450	8.95	8.50	8.00	*	*
4044 4Kx1 250	9.95	9.50	9.00	*	*
4045 1Kx4 450	8.95	8.50	8.00	*	*
4045 1Kx4 250	9.95	9.50	9.00	*	*
5257 low pow.	7.95	7.50	7.05	6.75	6.45

SPECIAL CIRCUITS

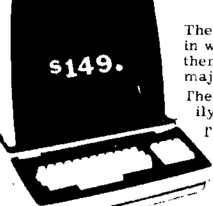
Z80A 4 MHz.	24.95	AY5-1013A UART	4.95
8080A CPU	9.95	Floppy Disc Controllers	
8085	22.50	WD 1771 single D.	39.95
8086 Intel 16 bits	*	WD 1781 Double D	65.00
TMS9900 16 bits	49.95	WD 1791 D/D 3740	*



EPROMS		1-15	16-63	64+
1702A	2K	4.95	4.50	4.00
2708	8K	9.95	9.50	9.00
2716	16K	19.95	*	*
2532	32K	*	*	*

CRT TERMINAL

Sanders Associates



$149.

These used vidio display terminals were in working condition when we purchased them from the reservation division of a major U.S. airline.
The terminals are RS-232 and should easily interface to most micro-computers.
To facilitate a quick liquidation of these units, we are offering them as-is, non-returnable for only $149.
Please include $35.00 for shipping, the balance will be refunded.

Thumbwheel switch

Ten position BCD $1.39 ea.

10	50
$1.19	.89

MINIATURE SWITCHES

your choice

	10	50	100	1k
$.98	$.88	.81	.73	.66

SPDT Miniature Toggles

7101 C&K	ON-NONE-ON	
7107 jbt	ON-OFF(mnt.ON)	
7108 CK	ON-(moment.ON)	
Rocker JBT	DPDT	
Rotary	3P-4-Pos.	
Rotary	3P-6-Pos.	
Push B (N.O.)	$.39ea.	4/$1

CAPACITORS

ELECTROLYTICS

	ea.	10	50
80,000/10v.	3.95	3.49	2.95
4500/50v.	$1.49	1.35	1.19
1000/15v axial	$.55	.49	.45
.1 disc	$.12	.09	.07
.01 disc	.06	.05	.04

DIP Switch

$1.49 ea.

10	25	100	1K
$1.29	1.15	.97	.83

specify 7 or 8 pos. SPST

50 Conductor RIBBON WIRE

$.69 Ft.

TWISTED PAIR

Power Adapter

6vdc, 140mA $1.39
7vdc, 1.4 A. 5.50
9vdc, 200mA. 1.19
10vAc, 300mA. 1.95

SPECIAL SPECIAL SPECIAL Only 10,000 Available

NE555H

Leads fan out to fit Mini-DIP socket.

3 for $.98

25 for $7.00 • 100 $19.50

Transistors

	ea.	10	50	100
2N2222A	.20	.18	.16	.15
2N3055	.69	.65	.59	.55
MJ3055	.79	.75	.69	.65
2N3772	1.59	1.49	1.39	1.29
2N3904	.15	.11	.09	.07
2N3906	.15	.11	.09	.07

Diodes

		10	25	100
1N4002	100v.	.08	06	.05
1N4005	600v.	.10	.08	.07
1N4148	signal	.07	.05	.04

jumbo red LED's

ea.	10	25	100
$.15	.13	.11	.09

$.98

Output: 12v.ct. 175mA. TRANSFORMER

TRIMMER POTENTIOMETERS

2K 5K 10K 50K

5 for $.98

20	50	100
16¢	14¢	12¢

9 foot $1.49

Heavy duty grounded power cord and mating chassis connectors.

Page Wire Wrap Kits precut & stripped

KIT No.1 $6.95 900 Assorted Lengths
KIT No.2 $19.95 2800 Assorted Lengths 250' Bulk Wire

DISCOUNT Wire Wrap Center

IC SOCKETS

pin	wire wrap ea.	25	50	low profile ea.	25	50
8				17¢	16	15
14	37¢	36	35	18	17	16
16	38	37	36	19	18	17
24	99	93	85	36	35	34
40	169	155	139	63	60	58

50ft. $.98 KYNAR WIRE WRAP

500	1,000	11,000
$9.	$15.	$105.

wire wrapping

$29.95 BW630

OK HOBBY WRAP-30 wire wrap & strip tool $5.45

VISA · master charge

(213) 679-9001

All merchandise sold by California Industrial is premium grade
Orders are shipped the same day received
PLEASE INCLUDE $1.00 SHIPPING ON ORDERS UNDER $15.00
California residents add 6% sales tax • Money back guarantee
Sorry, no COD's • Foreign orders add 10%
Credit cards accepted, $20 minimum

Circle 39 on inquiry card.

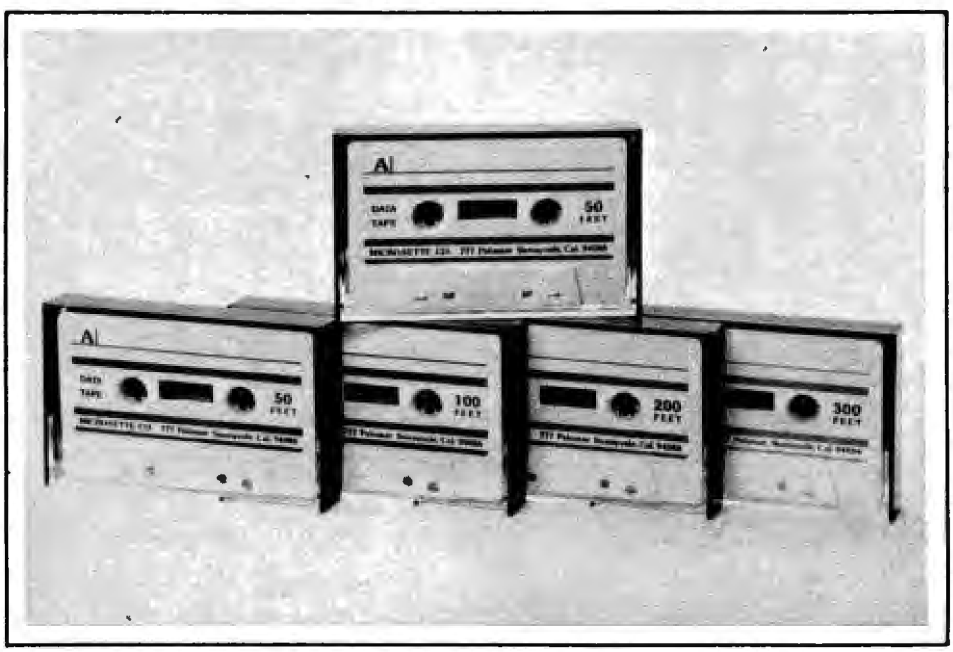
Mass Storage Unit Expands System 88 Filing Capabilities

PolyMorphic Systems has increased the storage capacities of its System 88 microcomputers through the introduction of a new option, the 88/MS, which consists of two drives for 8 inch floppy disks. One disk can hold 1.2 M bytes, or more than 500 pages of text. A System 88 microcomputer with one or two 88/MS units can handle all the files and processing needs of most small businesses and professional offices.

Present owners of any System 88 microcomputer can add the 88/MS mass storage unit with no changes in their equipment's operating system. Ready to use packages are available for doing tasks such as accounts receivable.

For more information on the 88/MS, contact PolyMorphic Systems, 460 Ward Dr, Santa Barbara CA 93111.■

Short Length Cassettes Designed for Personal Computers



Microsettes are short length, high quality cassettes designed for microcomputers. They feature a premium quality Philips cassette and high energy audio tape. Each cassette comes in a hard, 2 piece Norelco style box with two extra labels. The 50 foot length of tape in the C-10 Microsette provides slightly more than five minutes of recording per side. For additional information about the C-10 Microsette write to Microsette Co, 777 Palomar Av, Sunnyvale CA 94068.■

Circle 596 on inquiry card.

Floppy Disk Systems Software Transparent to RT-11

The new Remex-11 floppy disk systems are integrated hardware and software units that connect directly to the PDP-11 Unibus or LSI-11 Q bus. The systems are available with a utility function that permits data interchange between IBM 3740 diskettes and any RT-11 supported device.

The new plug compatible versions of the Remex-11, the Remex 11/11 and 11/12, are completely software transparent to the RT-11 software on the LSI-11 computers while offering added features. The Remex-11 provides read only memory bootstrapping as a standard feature as well as individual write protect switches to each drive, busy and error status indicators, and an automatic reinitialize function.

For increased performance while still maintaining media compatibility with PDP-11 and LSI-11 systems, the Remex 11/21 and 11/22 are available. These systems will accept up to four diskette drives. Data can be transferred in 16 bit words, and up to 65 K words can be transferred in a single input/output (IO) operation. For additions, the data buffer in the RT-11 controller can be increased to two full sectors.

The Remex 11/31 and 11/32 employ 16 sector and track soft sectoring format. A contiguous file allocation structure increases throughput by as much as 50 percent.

Both media compatible and expanded capacity systems connect to the PDP-11 by a bus extension cable; therefore no IO slot is required.

Remex-11 prices begin at $3195 complete. For further information contact Marketing Manager, Remex Division, Ex-Cell-O Corp, 1733 E Alton St, Irvine CA 92713.■

Circle 597 on inquiry card.

New Floppy Disks from Omni Products Company

New floppy disks are available from Omni Products Co, POB 223, Marlton NJ 08053. The disks include a full IBM compatible, soft sectored version, as well as Shugart compatible, hard sectored and Memorex compatible, hard sectored versions. They are designed to meet or exceed IBM and ANSI standards, and a written guarantee is furnished. Prices are $4.95 in quantities of one through nine and $4 for orders of more than 50. Include $1.50 for shipping per order. NJ residents should include 5% sales tax.■

Circle 598 on inquiry card.

16K E-PROM CARD

New!

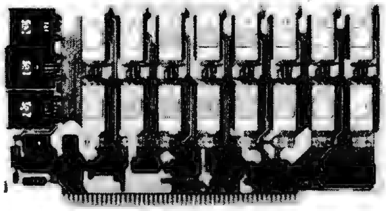
IMAGINE HAVING 16K OF SOFTWARE ON LINE AT ALL TIME!

S-100 (Imsai/Altair) Buss Compatible!

KIT FEATURES:

1. Double sided PC board with solder mask and silk screen and gold plated contact fingers.
2. Selectable wait states.
3. All address lines & data lines buffered!
4. All sockets included.
5. On card regulators.

KIT INCLUDES ALL PARTS AND SOCKETS (except 2708's). Add $25. for assembled and tested.



$59.95 kit

SPECIAL OFFER:

Our 2708's (450NS) are $8.95 when purchased with above kit.

NEW PRODUCTS FOR 1979

New Products are Scheduled for delivery during January 1979. Some may be available sooner. Call.

Z-80 CPU KIT
$129

For S-100 Buss. Features Jump on Reset capability. We feel this board has the most correct PSYNC signal for trouble free operation. Complete kit. More data on request. (for 4MHZ ADD $10)

16K STATIC RAM KIT
$295

For SS-50 (S.W. TECH. 6800) Buss. Fully static uses 2114 RAM's. 450 N.S. At last, a quality RAM board for this popular Buss at an affordable price. Complete kit. Additional Data on request.

DUAL DENSITY FLOPPY DISC CONTROLLER

For S-100 Buss. Reliability and Quality assured. Uses WD1791 Controller Chip. For 5¼ or 8 inch drives. Full IBM and S-100 compatible. Perfect for use with CP/M. Special P.L.L. design for maximum reliability. **Assembled & Tested - $219** Additional Data on Request.

CPM is a Trademark of Digital Research of California.

COMPUTER PARTS

Z-80	- $19.95	8080A	- $6.95
Z-80A	- 24.95	8212	- 2.25

ALARM CLOCK CHIP

N.S. MM5375AA. Six Digits. With full Data. **New!**

$2.49 each

FULL WAVE BRIDGE

4 AMP. 200 PIV.

69¢ 10 FOR $5.75

NOT ASSOCIATED WITH DIGITAL RESEARCH OF CALIFORNIA, THE SUPPLIERS OF CPM SOFTWARE.

MOTOROLA 7805R VOLTAGE REGULATOR

Same as standard 7805 except 750 MA output. TO—220. 5VDC output.

44c each or 10 for $3.95

450 NS! 2708 EPROMS

Now full speed! Prime new units from a major U.S. Mfg. 450 N.S. Access time. 1K x 8. Equiv. to 4-1702 A's in one package.

~~$15.75 ea.~~ PRICE CUT **$9.95** ~~4 FOR $50.00~~

16K STATIC RAM KIT

$295.00 COMPLETE KIT

ASSEMBLED & TESTED ADD $30

OUR LATEST COMPUTER KIT! FULLY S-100 COMPATIBLE! FULLY STATIC, AT DYNAMIC PRICES!

WHY THE 2114 RAM CHIP?

We feel the 2114 will be the next industry standard RAM chip (like the 2102 was). This means price, availability, and quality will all be good! Next, the 2114 is FULLY STATIC! We feel this is the **ONLY** way to go on the S-100 Buss! We've all heard the HORROR stories about some Dynamic Ram Boards having trouble with DMA and FLOPPY DISC DRIVES Who needs these kinds of problems? And finally, even among other 4K Static RAM's the 2114 stands out! Not all 4K static Rams are created equal! Some of the other 4K's have **clocked** chip enable lines and various timing windows **just as critical** as Dynamic RAM's. Some of our competitor's 16K boards use these "tricky" devices. But not us! The 2114 is the ONLY logical choice for a trouble-free, straightforward design

(450 NS)

BLANK PC BOARD WITH DOCUMENTATION - **$33.00**
SUPPORT IC'S + CAPS - **$19.95**
LOW PROFILE SOCKET SET - **$12.00**
2114'S 4K RAM'S - **8 FOR $69.95**

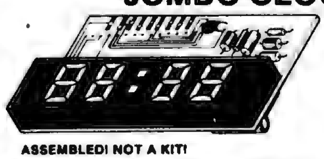
KIT FEATURES:

1. Addressable as four separate 4K Blocks.
2. ON BOARD BANK SELECT circuitry. (Cromemco Standard!) Allows up to 512K on line!
3. Uses 2114 (450NS) 4K Static Rams.
4. ON BOARD SELECTABLE WAIT STATES.
5. Double sided PC Board, with solder mask and silk screened layout. Gold plated contact fingers.
6. All address and data lines fully buffered.
7. Kit includes ALL parts and sockets.
8. PHANTOM is jumpered to PIN 67.
9. LOW POWER: under 2 amps TYPICAL from the +8 Volt Buss.
10. Blank PC Board can be populated as any multiple of 4K.

SUPER SPECIAL: BUY 32 KITS (512 KILOBYTES) (8-64K BANKS) for $8,995.00

NATIONAL SEMICONDUCTOR JUMBO CLOCK MODULE

MA1008A BRAND NEW!



$6.95

2 FOR $13

(AC XFMR $1.95)

FEATURES
- FOUR JUMBO ½ INCH LED DISPLAYS
- 12 HR REAL TIME FORMAT
- 24 HR ALARM SIGNAL OUTPUT
- 50 OR 60 Hz OPERATION
- LED BRIGHTNESS CONTROL
- POWER FAILURE INDICATOR
- SLEEP & SNOOZE TIMERS
- DIRECT LED DRIVE (LOW RFI)
- COMES WITH FULL DATA

ASSEMBLED! NOT A KIT!

ZULU VERSION!
We have a limited number of the 24 HR Real time version of this module in stock
#MA1008D — $9.95

PERFECT FOR USE WITH A TIMEBASE.

COMPARE AT UP TO TWICE OUR PRICE!

MANUFACTURER'S CLOSEOUT!

SALE!

1N4148 DIODES. SILICON. Same as 1N914. New, factory prime, Full Leads.
100 FOR $2
- 1000 FOR $17.50

New! REAL TIME Computer Clock Chip

N.S. MM5313. Features BOTH 7 segment and BCD outputs. 28 Pin DIP. **$4.95 with Data**

16K DYNAMIC RAM CHIP

16K X 1 Bits. 16 Pin Package Same as Mostek 4116-4. 250 NS access. 410 NS cycle time Our best price yet for this state of the art RAM. 32K and 64K RAM boards using this chip are readily available. These are new, fully guaranteed devices by a major mfg. **VERY LIMITED STOCK!**

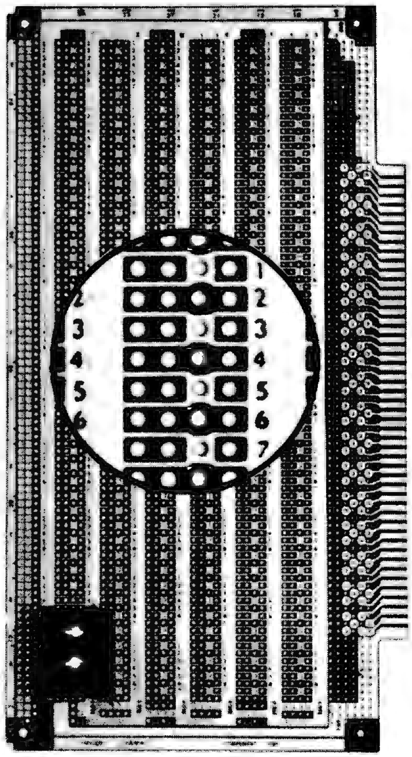
$14.95 each 8 FOR $89.95

Digital Research Corporation
(OF TEXAS)

P. O. BOX 401247Y /GARLAND, TEXAS 75040/(214) 271-2461

TERMS: Add 30¢ postage, we pay balance. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa, Mastercharge, and American Express cards. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H. 90 Day Money Back Guarantee on all items.

S-100 Bus Prototyping Circuit Board Accommodates up to 70 Integrated Circuits



This new prototyping circuit board is bus and shape compatible with Altair and IMSAI microcomputer boards. It holds up to 70 14 or 16 pin dual-in-line packages (DIP) or any combination of DIP sockets with 0.3, 0.4, 0.6 or 0.9 inch (0.76, 1.02, 1.52, 2.29 cm) lead spacing.

The 5.3 by 10 by 0.062 inch (13.46 by 25.4 by 0.16 cm) Model 8804 circuit board from Vector Electronics has 100 (50 each side) card edge contacts on 0.125 inch (0.32 cm) centers to accommodate an S-100 bus organization. Two 100 hole rows of individual 0.1 inch (0.25 cm) spaced pads across the top of the board permit additional input/output (IO) via ribbon wire assemblies. Wrap posts may be fabricated by inserting Vector's T46-5-9 wrapped wire pins into the holes.

For wiring convenience, power and ground buses are in an offset ladder pattern on opposite board sides. One corner of the 8804 may be used for a low profile heat sink with two regulators in the T0-220 packages. The leads of one regulator position are prewired to power input, ground and regulated power. The other regulator position is uncommitted.

The 8804 plug board is priced at $21.95 each in quantities of one to four. For further information, contact Vector Electronic Company, 12460 Gladstone Av, Sylmar CA 91342.■

Circle 606 on inquiry card.

Manual Details New Temperature Switch

An 8 page manual detailing the properties and applications of Midwest Components Inc's temperature switch is available from the company. This switch utilizes a reed switch and temperature activated magnets for sensing. For this manual, write to Midwest Components Inc, POB 787, 1981 Port City Blvd, Muskegon MI 49443.■

Circle 607 on inquiry card.

Paper on S-100 Bus Standards

A reprint from *Computer Magazine* entitled "Proposed Standard for the S-100 Bus," Preliminary Specification, IEEE Task 696.1/D2 was presented at the 1978 National Computer Conference. This document is a specification for both timing and signal disciplines and was prepared by two members of the IEEE Computer Society Microprocessor Standards Committee. It is available by sending a self-addressed, stamped envelope to Robert G Stewart, 1658 Belvoir Dr, Los Altos CA 94022.■

Intelligent Keyboard Has Capacitive Keyswitches

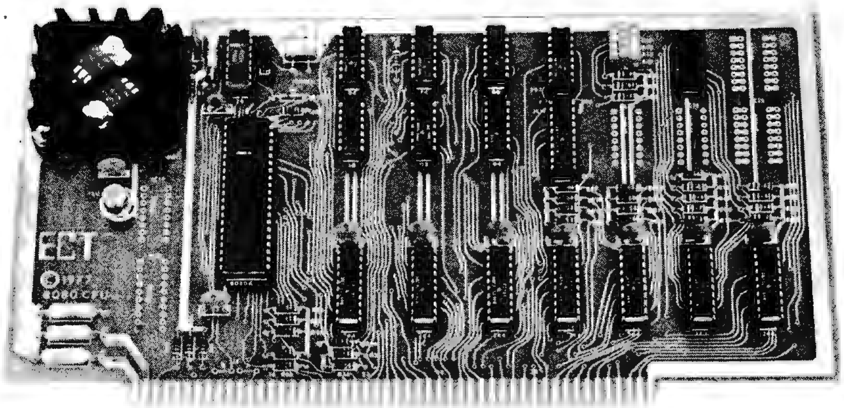
This new solid state keyboard uses a second generation microprocessor and low profile capacitive keyswitches. The keyboard uses an 8 bit single integrated circuit processor with on chip read only memory, programmable memory and erasable read only memory. All key functions are software controllable. The microprocessor permits automatic repeats, multiple application programs in a single intelligent encoder, field program changes using new firmware, serial and parallel input/output (IO), and n-key rollover (3 key rollover being standard).

The switches have a life expectancy of 100 million operations. The legends are selectable from a wide selection of symbols and letters in a host of languages and disciplines.

For further information about this keyboard, contact C P Clare & Company, 3101 W Pratt Av, Chicago IL 60645.■

Circle 609 on inquiry card.

8080 Processor Board Offered

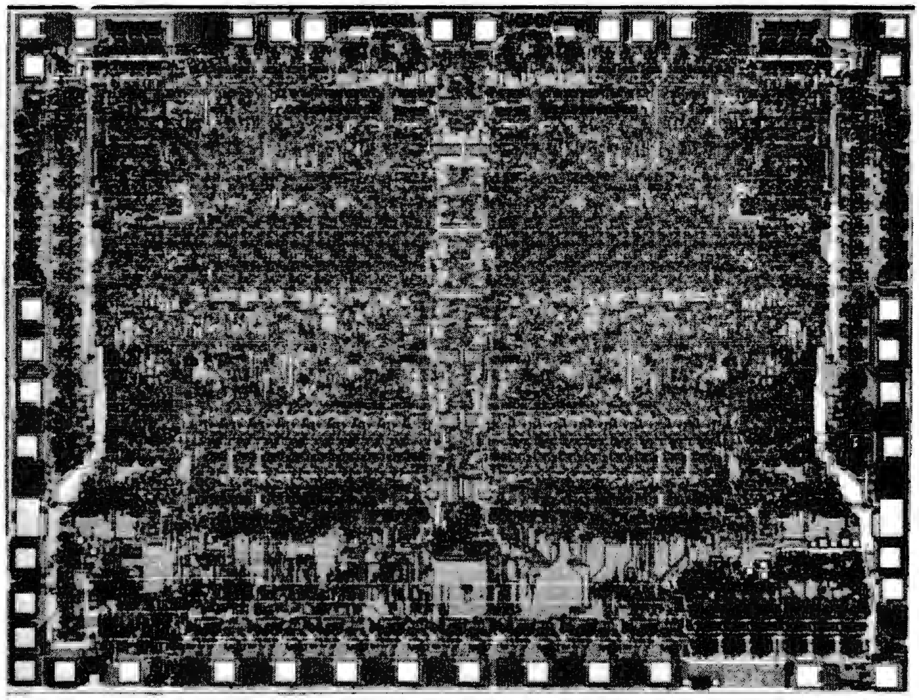


This S-100 bus 8 bit processor board uses the 8080A processor. 74LS244 bus drivers are utilized to provide low power with higher drive capability. A switch selectable jump on reset circuit is provided for use in systems without a front panel. Low profile sockets are provided for all integrated circuits.

The 8080 board is $175 assembled and $120 in kit form. For more information write to Electronic Control Technology, 763 Ramsey Av, Hillside NJ 07205.■

Circle 610 on inquiry card.

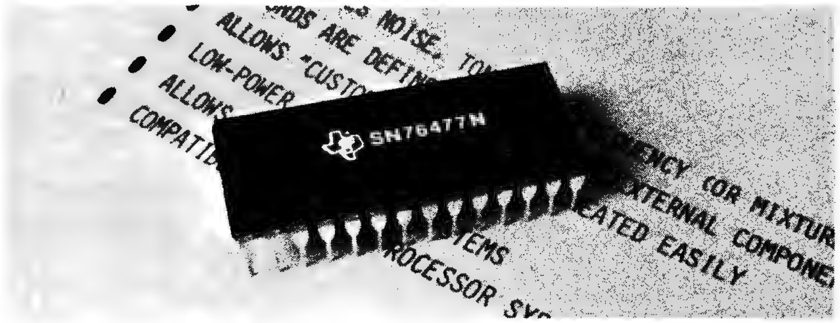
Single Chip Z80-SIO for LSI Microcomputer System



A high speed, dual channel, multi-protocol serial data communications controller circuit, the single chip Z80-SIO, has been introduced by Zilog, 10460 Bubb Road, Cupertino CA 95014. The SIO is designed to work with Zilog's Z-80 microcomputer family and also interfaces with most other 8 bit and 16 bit processors. The serial IO controller, an N/MOS 40 pin device, is a peripheral component that can control communications peripherals and format data in data communications networks. Each of the SIO's full duplex channels has four control lines for most commonly used modems. Applications include fiber optics, microwave transmission and satellite communications. For systems with 2.5 MHz clock rate, the SIO's data rate goes up to 550 K bits per second, while in a 4 MHz system, it's up to 880 K bits. Price is $49 in small quantities.■

Circle 622 on inquiry card.

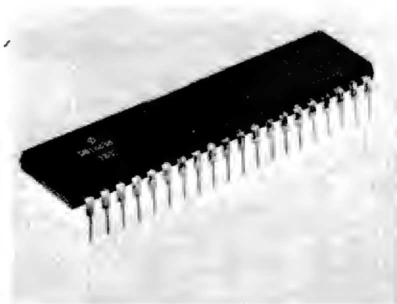
Complex Sound Generator Integrated Circuit



The SN76477N, a complex sound generator integrated circuit, has been announced by Texas Instruments, POB 84, M/S 812, Sherman TX 75090. This IC can be used to generate virtually any complex sound ie: siren, gunshot, jet engine, whistle, pinball sounds, etc. Since it is an integrated injection logic (I^2L) linear integrated circuit with low power consumption, it is ideally suited for battery powered applications. The SN76477N contains a voltage controlled oscillator, super low frequency oscillator, white noise generator, noise filter, oneshot, mixer, an attack and decay envelope generator. The desired sound is externally programmed by the user through logic and analog inputs. New sounds can be implemented or modified quickly. The SN76477N is designed for operation from −10 °C to 40 °C. It is offered in a 28 pin package. Price is $1.65 in quantities of 100.■

Circle 623 on inquiry card.

Single Channel DMAC from Western Digital

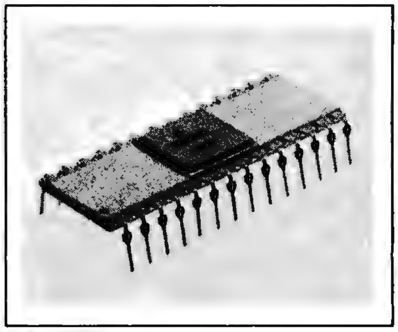


A low cost single channel direct memory access controller (DMAC) has been introduced by Western Digital Corp, 3128 Red Hill Av, POB 2180, Newport Beach CA 92663.

The DM 1883 is said to be fully compatible with all popular microprocessors built today. It includes the following features: control of all memory handshaking and device control; full 16 bit memory address and block length capability; block or word move; automatic end of block (EOB) shutoff and interrupt on EOB or error detection; and the option of auto load and bus timeout interrupt. The DM 1883 is powered by a single +5 V supply.■

Circle 624 on inquiry card.

Second Sourced TR 1953 USART Replaces 8251



The second sourced TR 1953 universal synchronous and asynchronous receiver transmitter (USART) is said to be the lowest priced replacement for the 8251.

The TR 1953's complete compatibility with the 8251 USART is further enhanced by synchronous and asynchronous operation, with 5 to 8 bit characters on both modes. Internal or external character synchronization and automatic sync insertion is featured in the synchronous mode; 1, 16 or 64 times bps rate, 1, 1½ or 2 stop bits and false start bit rejection on the asynchronous mode.

The TR 1953 comes in a 28 pin package, with TTL compatible IO, and operates on a single +5 V supply. Write to Western Digital Corp, 3128 Red Hill Av, POB 2180, Newport Beach CA 92663 for a sample.■

Circle 625 on inquiry card.

IN STOCK NOW

The EW-2001 A "Smart" VIDEO BOARD KIT At A "Dumb" Price!

A VIDEO BOARD + A MEMORY BOARD + AN I/O BOARD – ALL IN ONE!

■ STATE OF THE ART TECHNOLOGY USING DEDICATED MICROPROCESSOR I.C.
■ NUMBER OF I.C.s REDUCED BY 50% FOR HIGHER RELIABILITY ■ MASTER PIECE OF ENGINEERING ■ FULLY SOFTWARE CONTROLLED

Priced at ONLY **$199.95** Basic Software Included

SPECIAL FEATURES:

- S-100 bus compatible
- Parallel keyboard port
- On board 4K screen memory (optional)* relocatable to main computer memory
- Text editing capabilities (software optional)
- Scrolling: up and down through video memory
- Blinking characters
- Reversed video
- Provision for on board ROM
- CRT and video controls fully programmable (European TV)
- Programmable no. of scan lines
- Underline blinking cursor
- Cursor controls: up, down, left, right, home, carriage return
- Composite video

*Min. 2K required for operation of this board.

DISPLAY FEATURES:

- 128 displayable ASCII characters (upper and lower case alphanumeric, controls)
- 64 or 32 characters per line (jumper selectable)
- 32 or 16 lines (jumper selectable)
- Screen capacity 2048 or 512
- Character generation: 7 x 11 dot matrix

OPTIONS:

Sockets	$10.00
2K Static Memory (with Sockets)	$45.00
4K Static Memory (with Sockets)	$90.00
Complete unit, assembled and tested with 4K Memory	$335.00
Basic software on ROM	$20.00
Text editor on ROM	$75.00

DEALER INQUIRIES WELCOMED

8080 SUPPORT

8212	$3.00
8214	7.95
8216	3.50
8228	5.95
8251	7.95
8255	8.50

8080A CPU $7.75

RAM-2114 1Kx4 450ns $8.00

GROUNDED A.C. CORDS 6 Ft. – $1.19

WIRE WRAP WIRE KYNAR 28 AWG $2.95/100 Feet Blue, Red, White

Computer Grade Capacitors 5000 mfd 60VDC $1.50

EDGE CONNECTOR WIRE WRAP PINS
44 Pin $1.25 – 72 Pin $1.75 – 100 Pin (S-100) $5.45

MISC. IC's

DM8810	3/$1
DM8210	$2 ea.
N8T15	$1 ea.
9024	2/$1.50
93L08	$1.50 ea.
93L09	2/$1.50
93L24	$1 ea.

CMOS

4011	6/$1.00
4022	2/$1.70
4023	4/$1.00
4071	5/$1.00

LINEAR I.C.'s

LM320K-5	.99
LM320-12	.99
LM709N	11/$1
LM710H	6/$1
LM711H	6/$1
LM741M	7/$1

Transistors & Diodes

2N3906	7/$1
2N4400	9/$1
2N4403	9/1
2N2222A	4/$1
1N4003	12/$1
1N4005	10/$1
1N4148	14/$1
SCR 400V 4A, TO220	$ .80

SHIPPING: Keyboard and Video Board: $3.50; OTHERS 1.25

California residents add 6% sales tax

ELECTRONICS WAREHOUSE Inc.
1603 AVIATION BLVD.
REDONDO BEACH, CA. 90278
TEL. (213) 376-8005
WRITE FOR FREE CATALOG
Minimum Order: $10

ASCII KEYBOARD KIT $74.00

Additional Improvements: Double Size Return Key
Control Characters Molderd on Key Caps

- Power: +5V 275mA
- Upper and Lower Case
- Full ASCII Set
- 7 or 8 Bits Parallel Data
- Optional Serial Output
- Selectable Positve or Negative Strobe, and Strobe Pulse Width
- 2 Key Roll-Over
- 3 User DEfineable Keys
- P.C. Board Size: 17-3/16" x 5"

OPTIONS:

Metal Enclosure Painted Blue and White	$27.50
18 Pin Edge Con.	$ 2.00
I.C. Sockets	$ 4.00
Serial Output Provision (Shift Register)	$ 2.00
Upper Case Lock Switch for Capital Letters and Nos.	$ 2.00
Assembled (on Sockets) and Tested	$90.00

APPLE II I/O BOARD KIT

Plugs Into Slot of Apple II Mother Board

18 Bit Parallel Output Port (Expandable to 3 Ports)
1 Input Port
15mA Output Current Sink or Source
Can be used for peripheral equipment such as printers, floppy discs, cassettes, paper tapes, etc.

1 free software listing for SWTP PR40 or IBM selectric

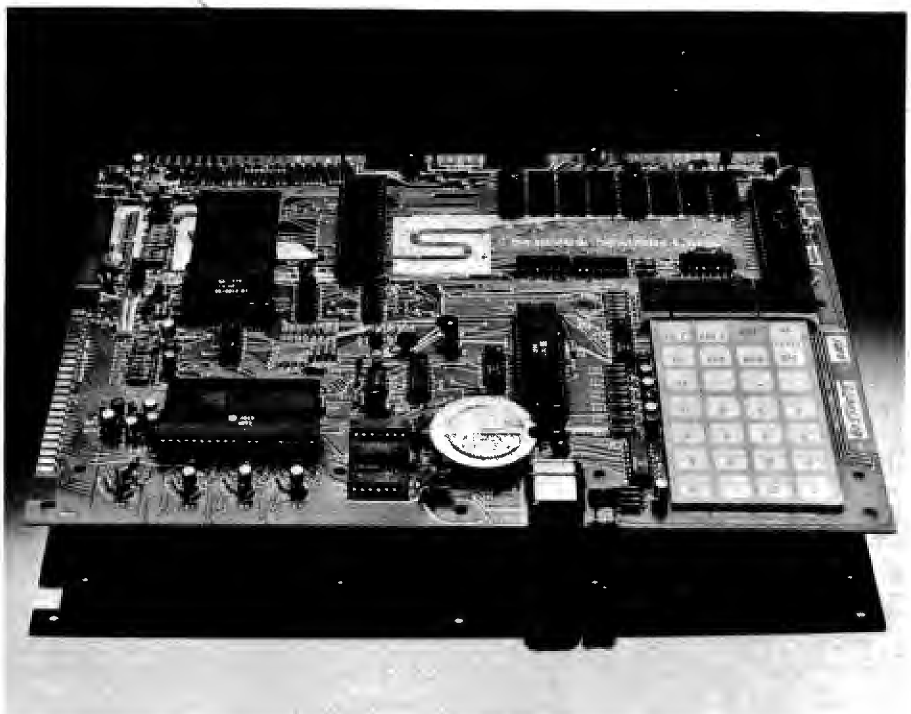
PRICE:
1 Input and 1 Output Port for $49.00
1 Input and 3 Output Ports for $64.00
Dealer Inquiries Invited

Circle 130 on inquiry card.

What's New?

SYSTEMS

A Single Board Microcomputer System



The SYM-1 is a complete microcomputer including keyboard, display and operating software. Some of the features of this stand-alone system include a SY6502 microprocessor device; 1 K bytes of user programmable memory, expandable to 4 K bytes in board sockets; 28 key audio response keypad; RS-232 and current loop interfaces allow operation of the system with video or Teletype facilities; 4 K byte read only memory system monitor; 25 general purpose input/output (IO) lines; SUPERMON software system; hardware compatible with KIM board; and more. Other hardware features include oscilloscope output display and four general purpose IO buffers.

The SYM-1 is priced at $269. Contact Synertek, 3050 Coronado Dr, Santa Clara CA 95051.■

Circle 626 on inquiry card.

New Microcomputer System from OSI

Ohio Scientific has announced a new microcomputer system, designated the C3-OEM. Its applications include use as a general purpose computer or controller in large equipment such as medical diagnostic equipment, scientific equipment, analytical equipment and industrial control applications. It is also suited for small systems software since it will run software for the 6502, 6800, 8080 or Z-80.

Its features include: single chassis construction, which can be either tabletop mounted or rack mounted, including dual 8 inch floppy disks for 500 K bytes of on line storage, 32 K bytes of static programmable memory, one RS-232 port, and Ohio Scientific's triple processor board which supports the 6502A, 6800 and Z-80 processors. The system comes complete with a 6502 disk operating system and BASIC for disk and multiple processor switching software. The unit features modular construction with an 8 slot mother board, of which only four slots are used in the base machine. Additional options include: additional memory, a 16 serial port input and output (IO) board and a 96 line parallel IO board.

Single unit price is $3590. Contact Ohio Scientific, 1333 S Chillicothe Rd, Aurora OH 44202.■

Circle 628 on inquiry card.

A System for Data Handling

The VP (video processor) series is a complete computer system with standard interfaces for expandability. The basic unit includes: video monitor, encoded keyboard, minifloppy disk drive, 32 K bytes of programmable memory, serial RS-232 port, four counter timers, printer port, 16 programmable IO lines, and a Z-80 processor. Interface electronics and card cage are available for use with S-100 boards. The main processor board allows expansion to 48 K bytes of programmable memory, 8 K bytes of erasable read only memory, four additional counter timers, and 48 programmable IO lines. The disk controller works with either 8 inch (20.32 cm) or 5.25 inch (13.34 cm) floppy drives and supports three additional drives.

All VP series units include a video display offering programmable screen formats with up to 80 characters per line and 24 lines per frame. The video processor has graphic capabilities and supports an optional light pen. Reverse video, blinking and highlight for single characters or fields are included, as well as an underline or block cursor.

The entire VP series includes the CP/M disk operating system. Also available are several BASIC interpreters and the C-BASIC compiler. Text editing and assembler are included with the software.

The smallest system, VP-80, includes 32 K bytes of programmable memory. It is available to dealers and educational facilities for $3995. All units are assembled and tested. Contact Data World Inc, 7541 Ravensridge Dr, St Louis MO 63119.■

Circle 627 on inquiry card.

Low Priced Microprocessor Comes Completely Assembled

This complete ready to use microprocessor offers an economical solution for both scientific applications and industrial usage. The MICRO-68 computer system is priced at $495 and comes completely assembled. Built around the Motorola/AMI/Hitachi 6800 processor, the MICRO-68 comes with its own integral power supply, 16 button keyboard, 6 digit LED display, and 128 words of programmable memory. The 512 MON-1 Bug programmable read only memory contains all the service necessary to load programs easily, inspect and edit them as necessary, insert break points for debugging, and execute. Memory expansion to 64 K bytes and full 16 bit input and output can be obtained via the edge connectors, which are provided for. All of the memory lines of the MICRO-68 can be buffered on board. The MICRO-68 comes in a hardwood cabinet with a transparent smoked Plexiglas lid. The unit measures 9 by 16 by 2 inches (22.86 by 40.64 by 5.08 cm). Contact EPA Electronic Product Associates Inc, 1157 Vega St, San Diego CA 92110.■

Circle 629 on inquiry card.

DIODES/ZENERS

1N914	100v	10mA	.05
1N4005	600v	1A	.08
1N4007	1000v	1A	.15
1N4148	75v	10mA	.05
1N4733	5.1v	1 W Zener	.25
1N753A	6.2v	500 mW Zener	.25
1N758A	10v	"	.25
1N759A	12v	"	.25
1N5243	13v	"	.25
1N5244B	14v	"	.25
1N5245B	15v	"	.25

SOCKETS/BRIDGES

8-pin	pcb	.20	ww	.35
14-pin	pcb	.20	ww	.40
16-pin	pcb	.20	ww	.40
18-pin	pcb	.25	ww	.75
22-pin	pcb	.35	ww	.95
24-pin	pcb	.35	ww	.95
28-pin	pcb	.45	ww	1.25
40-pin	pcb	.50	ww	1.25

Molex pins	.01	To-3 Sockets	.25
2 Amp Bridge		100-prv	.95
25 Amp Bridge		200-prv	1.95

TRANSISTORS, LEDS, etc.

2N2222	NPN	(2N2222 Plastic .10)	.15
2N2907	PNP		.15
2N3906	PNP	(Plastic - Unmarked)	.10
2N3904	NPN	(Plastic - Unmarked)	.10
2N3054	NPN		.35
2N3055	NPN	15A 60v	.50
T1P125	PNP	Darlington	.95
LED Green, Red, Clear, Yellow			.15
D.L.747		7 seg 5/8" High com-anode	1.95
MAN72		7 seg com-anode (Red)	1.25
MAN3610		7 seg com-anode (Orange)	1.25
MAN82A		7 seg com-anode (Yellow)	1.25
MAN74A		7 seg com-cathode (Red)	1.50
FND359		7 seg com-cathode (Red)	1.25

C MOS

4000	.15
4001	.15
4002	.20
4004	3.95
4006	.95
4007	.20
4008	.75
4009	.35
4010	.35
4011	.20
4012	.20
4013	.40
4014	.75
4015	.75
4016	.35
4017	.75
4018	.75
4019	.35
4020	.85
4021	.75
4022	.75
4023	.20
4024	.75
4025	.20
4026	1.95
4027	.35
4028	.75
4030	.35
4033	1.50
4034	2.45
4035	.75
4040	.75
4041	.69
4042	.65
4043	.50
4044	.65
4046	1.25
4049	.45
4050	.45
4066	.55
4069/74C04	.25
4071	.25
4081	.30
4082	.30
MC 14409	14.50
MC 14419	4.85
4511	.95
74C151	1.90

– T T L –

7400	.10	7473	.25	74176	.85	74H72	.35	74S133	.40
7401	.15	7474	.30	74180	.55	74H101	.75	74S140	.55
7402	.15	7475	.35	74181	2.25	74H103	.55	74S151	.30
7403	.15	7476	.40	74182	.75	74H106	.95	74S153	.35
7404	.10	7480	.55	74190	1.25			74S157	.75
7405	.25	7481	.75	74191	.95	74L00	.25	74S158	.30
7406	.25	7483	.75	74192	.75	74L02	.20	74S194	1.05
7407	.55	7485	.55	74193	.85	74L03	.25	74S257 (8123)	1.05
7408	.15	7486	.25	74194	.95	74L04	.30		
7409	.15	7489	1.05	74195	.95	74L10	.20	74LS00	.20
7410	.15	7490	.45	74196	.95	74L20	.35	74LS01	.20
7411	.25	7491	.70	74197	.95	74L30	.45	74LS02	.20
7412	.25	7492	.45	74198	1.45	74L47	1.95	74LS04	.20
7413	.25	7493	.35	74221	1.00	74L51	.45	74LS05	.25
7414	.75	7494	.75	74367	.75	74L55	.65	74LS08	.25
7416	.25	7495	.60			74L72	.45	74LS09	.25
7417	.40	7496	.80	75108A	.35	74L73	.40	74LS10	.25
7420	.15	74100	1.15	75491	.50	74L74	.45	74LS11	.25
7426	.25	74107	.25	75492	.50	74L75	.55	74LS20	.20
7427	.25	74121	.35			74L93	.55	74LS21	.25
7430	.15	74122	.55			74L123	.85	74LS22	.25
7432	.20	74123	.35	74H00	.15			74LS32	.25
7437	.20	74125	.45	74H01	.20	74S00	.35	74LS37	.25
7438	.20	74126	.35	74H04	.20	74S02	.35	74LS38	.35
7440	.20	74132	.75	74H05	.20	74S03	.25	74LS40	.30
7441	1.15	74141	.90	74H08	.35	74S04	.25	74LS42	.65
7442	.45	74150	.85	74H10	.35	74S05	.35	74LS51	.35
7443	.45	74151	.65	74H11	.25	74S08	.35	74LS74	.35
7444	.45	74153	.75	74H15	.45	74S10	.35	74LS86	.35
7445	.65	74154	.95	74H20	.25	74S11	.35	74LS90	.55
7446	.70	74156	.70	74H21	.25	74S20	.25	74LS93	.55
7447	.70	74157	.65	74H22	.40	74S40	.20	74LS107	.40
7448	.50	74161	.55	74H30	.20	74S50	.20	74LS123	1.00
7450	.25	74163	.85	74H40	.25	74S51	.25	74LS151	.75
7451	.25	74164	.60	74H50	.25	74S64	.15	74LS153	.75
7453	.20	74165	1.10	74H51	.25	74S74	.35	74LS157	.75
7454	.25	74166	1.25	74H52	.15	74S112	.60	74LS164	1.00
7460	.40	74175	.80	74H53J	.25	74S114	.65	74LS193	.95
7470	.45			74H55	.20			74LS367	.75
7472	.40							74LS368	.65

LINEARS, REGULATORS, etc.

MCT2	.95	LM320T5	1.65	LM340K15	1.25	LM723	.40
8038	3.95	LM320T12	1.65	LM340K18	1.25	LM725N	2.50
LM201	.75	LM320T15	1.65	LM340K24	1.25	LM739	1.50
LM301	.45	LM324N	1.25	78L05	.75	LM741 (8-14)	.25
LM308 (Mini)	.95	LM339	.75	78L12	.75	LM747	1.10
LM309H	.65	7805 (340T5)	.95	78L15	.75	LM1307	1.25
LM309K (340K-5)	.85	LM340T12	.95	78M05	.75	LM1458	.65
LM310	.85	LM340T15	.95	LM373	2.95	LM3900	.50
LM311D (Mini)	.75	LM340T18	.95	LM380 (8-14 PIN)	.95	LM75451	.65
LM318 (Mini)	1.75	LM340T24	.95	LM709 (8, 14 PIN)	.25	NE555	.35
LM320K5 (7905)	1.65	LM340K12	1.25	LM711	.45	NE556	.85
LM320K12	1.65					NE565	.95
						NE566	1.25
						NE567	.95

9000 SERIES

9301	.85	95H03	1.10
9309	.35	9601	.20
9322	.65	9602	.45

MICRO'S, RAMS, CPU'S, E-PROMS

74S188	3.00	8214	8.95
1702A	4.50	8224	3.25
MM5314	3.00	8228	6.00
MM5316	3.50	8251	8.50
2102-1	1.45	8255	8.50
2102L-1	1.75	8T13	1.50
2114	9.50	8T23	1.50
TR1602B	3.95	8T24	2.00
TMS 4044-	9.95	8T97	1.00
		2107B-4	4.95
8080	8.95	2708	9.50
8212	2.95	Z80 PIO	8.50

INTEGRATED CIRCUITS UNLIMITED

7889 Clairemont Mesa Boulevard, San Diego, California 92111
(714) 278-4394 *(Calif. Res.)*

All orders shipped prepaid — No minimum
Open accounts invited — COD orders accepted

Discounts available at OEM Quantities — California Residents add 6% Sales Tax
All IC's Prime/Guaranteed. All orders shipped same day received.

24 Hour Toll Free Phone 1-800-854-2211 — **American Express / BankAmericard / Visa / MasterCharge**

SPECIAL DISCOUNTS

Total Order	Deduct
$35 - $99	10%
$100 - $300	15%
$301 - $1000	20%

Circle 180 on inquiry card.

7400 TTL

SN7400N	.16	SN7470N	.29	SN74160N	.89
SN7401N	.18	SN7472N	.29	SN74161N	.89
SN7402N	.18	SN7473N	.35	SN74162N	1.95
SN7403N	.18	SN7474N	.35	SN74163N	.89
SN7404N	.18	SN7475N	.49	SN74164N	.89
SN7405N	.20	SN7476N	.35	SN74165N	.89
SN7406N	.29	SN7479N	5.00	SN74166N	1.25
SN7407N	.29	SN7480N	.50	SN74167N	1.95
SN7408N	.20	SN7482N	.99	SN74170N	1.59
SN7409N	.20	SN7483N	.59	SN74172N	6.00
SN7410N	.18	SN7485N	.79	SN74173N	1.25
SN7411N	.25	SN7486N	.35	SN74174N	.89
SN7412N	.25	SN7489N	1.75	SN74175N	.79
SN7413N	.40	SN7490N	.45	SN74176N	.79
SN7414N	.70	SN7491N	.59	SN74177N	.79
SN7416N	.25	SN7492N	.43	SN74179N	1.95
SN7417N	.25	SN7493N	.43	SN74180N	.79
SN7420N	.20	SN7494N	.65	SN74181N	1.95
SN7421N	.29	SN7495N	.65	SN74182N	.79
SN7422N	.39	SN7496N	.65	SN74184N	1.95
SN7423N	.25	SN7497N	3.00	SN74185N	1.95
SN7425N	.29	SN74100N	.89	SN74186N	9.95
SN7426N	.29	SN74107N	.35	SN74188N	3.95
SN7427N	.25	SN74109N	.59	SN74190N	1.25
SN7429N	.39	SN74116N	1.95	SN74191N	1.25
SN7430N	.20	SN74121N	.35	SN74192N	.79
SN7432N	.25	SN74122N	.39	SN74193N	.79
SN7437N	.25	SN74123N	.49	SN74194N	.89
SN7438N	.25	SN74125N	.49	SN74195N	.69
SN7439N	.25	SN74126N	.49	SN74196N	.89
SN7440N	.20	SN74132N	.75	SN74197N	.89
SN7441N	.89	SN74136N	.75	SN74198N	1.49
SN7442N	.49	SN74141N	.79	SN74199N	1.49
SN7443N	.75	SN74142N	2.95	SN74S200	4.95
SN7444N	.75	SN74143N	2.95	SN74251N	1.79
SN7445N	.75	SN74144N	2.95	SN74279N	.79
SN7446N	.69	SN74145N	.79	SN74283N	2.25
SN7447N	.59	SN74147N	1.95	SN74284N	3.95
SN7448N	.79	SN74148N	1.29	SN74285N	3.95
SN7450N	.20	SN74150N	.89	SN74365N	.69
SN7451N	.20	SN74151N	.59	SN74366N	.69
SN7453N	.20	SN74152N	.59	SN74367N	.69
SN7454N	.20	SN74153N	.59	SN74368N	.69
SN7459A	.25	SN74154N	.99	SN74390N	1.95
SN7460N	.20	SN74155N	.79	SN74393N	1.95
		SN74156N	.79		
		SN74157N	.65		

20% Discount 100 pcs combined order 25% -1000 pcs combined order

C/MOS

CD4000	.23	CD4028	.89	CD4070	.55
CD4001	.23	CD4029	1.19	CD4071	.23
CD4002	.23	CD4030	.49	CD4072	.49
CD4006	1.19	CD4035	.99	CD4076	1.39
CD4007	.25	CD4040	1.19	CD4081	.23
CD4009	.49	CD4041	1.25	CD4082	.23
CD4010	.49	CD4042	.99	CD4093	.99
CD4011	.23	CD4043	.89	CD4098	2.49
CD4012	.25	CD4044	.89	MC14409	14.95
CD4013	.39	CD4046	1.79	MC14410	14.95
CD4014	1.39	CD4047	2.50	MC14411	14.95
CD4015	1.19	CD4048	1.35	MC14419	4.95
CD4016	.49	CD4049	.49	MC14433	19.95
CD4017	1.19	CD4050	.49	MC14506	.75
CD4018	.99	CD4051	1.19	MC14507	.99
CD4019	.49	CD4053	1.19	MC14562	14.50
CD4020	1.19	CD4056	2.95	MC14583	3.50
CD4021	1.39	CD4059	9.95	CD4508	3.95
CD4022	1.19	CD4060	1.49	CD4510	1.39
CD4023	.23	CD4066	.79	CD4511	1.29
CD4024	.79	CD4068	.39	CD4515	2.95
CD4025	.23	CD4069	.45	CD4518	1.29
CD4026	2.25			CD4520	1.29
CD4027	.69			CD4566	2.25

74C00

74C00	.39	74C85	2.49	74C163	2.49
74C02	.39	74C90	1.95	74C164	2.49
74C04	.39	74C93	1.95	74C173	2.60
74C08	.49	74C95	1.95	74C192	2.49
74C10	.39	74C107	1.25	74C193	2.49
74C14	1.95	74C151	2.90	74C195	2.49
74C20	.39	74C154	3.00	74C922	5.95
74C30	.39	74C157	2.15	74C923	6.25
74C42	1.95	74C160	2.49	74C925	8.95
74C48	2.49	74C161	2.49	74C926	8.95
74C73	.89			80C95	1.50
74C74	.89			80C97	1.50

LINEAR

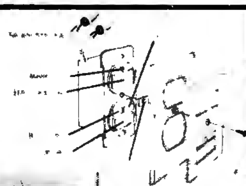
78MG	1.75	LM340K-18	1.35	LM710N	.79
LM105H	.99	LM340K-24	1.35	LM711N	.39
LM300H	.80	LM340T-5	1.25	LM723N/H	.55
LM301CN/H	.35	LM340T-6	1.25	LM733N	1.00
LM302H	.75	LM340T-8	1.25	LM739N	1.19
LM304H	1.00	LM340T-12	1.25	LM741CN/H	.35
LM305H	.60	LM340T-15	1.25	LM741-14N	.39
LM307CN/H	.35	LM340T-18	1.25	LM747N/H	.79
LM308CN/H	1.00	LM340T-24	1.25	LM748N/H	.39
LM309H	1.10	LM358N	1.00	LM1310N	2.95
LM309K	1.25	LM370N	1.95	LM1458CN/H	.59
LM310CN	1.15	LM373N	3.25	MC1488N	1.95
LM311N/H	.90	LM377N	4.00	MC1489N	1.95
LM312H	1.95	LM380N	1.25	LM1496N	.95
LM317K	6.50	LM380CN	.99	LM1556V	1.75
LM318CN/H	1.50	LM381N	1.79	MC1741SCP	3.00
LM319N	1.30	LM382N	1.79	LM2111N	1.95
LM320K-5	1.35	NE501N	8.00	LM2901N	2.95
LM320K-5.2	1.35	NE510A	6.00	LM3053N	1.50
LM320K-12	1.35	NE529A	4.95	LM3065N	1.49
LM320K-15	1.35	NE531H/V	3.95	LM3900N(3401)	.49
LM320K-18	1.35	NE536T	6.00	LM3905N	.89
LM320K-24	1.35	NE540L	6.00	LM3909N	1.25
LM320T-5	1.25	NE544N	4.95	MC5558V	.59
LM320T-5.2	1.25	NE550N	1.30	8038B	4.95
LM320T-8	1.25	NE555V	.39	LM75450N	.49
LM320T-12	1.25	NE556N	.99	75451CN	.39
LM320T-15	1.25	NE560B	5.00	75452CN	.39
LM320T-18	1.25	NE561B	5.00	75453CN	.39
LM320T-24	1.25	NE562B	5.00	75454CN	.39
LM323K-5	5.95	NE565N/H	1.25	75491CN	.79
LM324N	1.80	NE566CN	1.75	75492CN	.89
LM339N	.99	NE567V/H	.99	75493N	.89
LM340K-5	1.35	NE570N	4.95	75494CN	.89
LM340K-6	1.35	LM703CN/H	.69	RC4136	1.25
LM340K-8	1.35	LM709N/H	.29	RC4151	5.95
LM340K-12	1.35			RC4194	5.95
LM340K-15	1.35			RC4195	4.49

74LS00TTL

74LS00	.23	74LS47	.69	74LS138	.69
74LS01	.23	74LS51	.23	74LS139	.69
74LS02	.23	74LS54	.23	74LS151	.69
74LS03	.23	74LS55	.23	74LS155	.69
74LS04	.29	74LS73	.35	74LS157	.69
74LS05	.29	74LS74	.35	74LS160	.89
74LS08	.29	74LS75	.49	74LS161	.89
74LS09	.29	74LS76	.35	74LS162	.89
74LS10	.23	74LS78	.39	74LS163	.89
74LS11	.60	74LS83	.75	74LS164	.99
74LS13	.49	74LS85	.99	74LS175	.89
74LS14	.99	74LS86	.35	74LS181	2.49
74LS15	.29	74LS90	.49	74LS190	.89
74LS20	.23	74LS92	.59	74LS191	.89
74LS21	.29	74LS93	.59	74LS192	.89
74LS22	.29	74LS95	.79	74LS193	.89
74LS26	.29	74LS96	.89	74LS194	.89
74LS27	.29	74LS107	.35	74LS195	.89
74LS28	.29	74LS109	.35	74LS253	.79
74LS30	.23	74LS112	.35	74LS257	.69
74LS32	.29	74LS123	.99	74LS258	1.39
74LS37	.35	74LS125	.69	74LS260	.55
74LS40	.29	74LS132	.79	74LS279	.59
74LS42	.69	74LS136	.39	74LS367	.59
				74LS368	.59
				74LS670	1.95

Vigilite

Electronic Home Security Timer



A microprocessor-based pre-programmed light control that fits into a home wall socket, replacing the standard on/off light switch. Discourages intruders and burglars by turning lights on and off in a "real-life" pattern while you're away.

Unlike other electromechanical timers Vigilite can simulate the lighting patterns of five different rooms as selected by the user. Vigilite can also control over-head lights, which other timers cannot. Three Vigilite units, simulating kitchen, bathroom, and bedroom lighting can give a home ultimate protection, because the user chooses a lighting pattern that depicts his real life pattern. He then sets the Vigilite clock and room pattern accordingly. (See bar chart below.) The house actually looks occupied, although no one is home.

Easy to install, the Vigilite unit contains an accurate digital LED clock, plus pre-programmed independent lighting patterns for bedroom, bathroom, kitchen, living room, and outside porch lights. All-solid-state components assure the user long product life and reliability.

Part Number VGL-1 $39.95 ea.

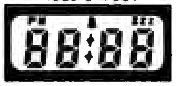
Technical Specifications

1) Electrical requirement—120 VAC, 60Hz, 2½ Amps.
2) For use with permanently installed incandescent 40 W to 300 W light fixtures—single pole circuits.
3) Cannot be used to control television, appliances, or other motor driven equipment
4) One vigilite unit required for each room circuit

DISCRETE LEDS

.200" dia.			**.125" dia.**		
XC556R	red	5/$1	XC209R	red	5/$1
XC556G	green	4/$1	XC209G	green	4/$1
XC556Y	yellow	4/$1	XC209Y	yellow	4/$1
XC556C	clear	4/$1			
.200" dia.			**.185" dia.**		
XC22R	red	5/$1	XC526R	red	5/$1
XC22G	green	4/$1	XC526G	green	4/$1
XC22Y	yellow	4/$1	XC526Y	yellow	4/$1
.170" dia.			XC526C	clear	4/$1
MV10B	red	4/$1	**.190" dia.**		
.085" dia.			XC111R	red	5/$1
MV50	red	6/$1	XC111G	green	4/$1
INFRA-RED LED 1/4"x1/4"x1/16" flat		5/$1	XC111Y	yellow	4/$1
			XC111C	clear	4/$1

TIMEX T1001
LIQUID CRYSTAL DISPLAY
FIELD EFFECT



4 DIGIT – .5" CHARACTERS
THREE ENUNCIATORS
2.00" X 1.20" PACKAGE
INCLUDES CONNECTOR

T1001-Transmissive $7.95
T1001A-Reflective 8.25

DISPLAY LEDS

TYPE	POLARITY	HT	PRICE	TYPE	POLARITY	HT	PRICE
MAN 1	Common Anode-red	.270	2.95	MAN 6730	Common Anode-red ± 1	.560	.99
MAN 2	5 x 7 Dot Matrix-red	.300	4.95	MAN 6740	Common Cathode-red-D D	.560	.99
MAN 3	Common Cathode-red	.125	.25	MAN 6750	Common Cathode-red ± 1	.560	.99
MAN 4	Common Cathode-red	.187	1.95	MAN 6760	Common Anode-red	.560	.99
MAN 7G	Common Anode-green	.300	1.25	MAN 6780	Common Cathode-red	.560	.99
MAN 7Y	Common Anode-yellow	.300	.99	DL701	Common Anode-red ± 1	.300	.99
MAN 72	Common Anode-red	.300	.99	DL704	Common Cathode-red	.300	.99
MAN 74	Common Cathode-red	.300	1.25	DL707	Common Anode-red	.300	.99
MAN 82	Common Anode-yellow	.300	.99	DL728	Common Cathode-red	.500	1.49
MAN 84	Common Cathode-yellow	.300	.99	DL741	Common Anode-red	.600	1.25
MAN 3620	Common Anode-orange	.300	.99	DL746	Common Anode-red ± 1	.630	1.49
MAN 3630	Common Anode-orange ± 1	.300	.99	DL747	Common Anode-red	.600	1.49
MAN 3640	Common Cathode-orange	.300	.99	DL749	Common Cathode-red ± 1	.630	1.49
MAN 4610	Common Anode-orange	.400	.99	DL750	Common Cathode-red	.600	1.49
MAN 4640	Common Cathode-orange	.400	.99	DL338	Common Cathode-red	.110	.35
MAN 4710	Common Anode-red	.400	.99	FND70	Common Cathode	.250	.69
MAN 4730	Common Anode-red ± 1	.400	.99	FND358	Common Cathode ± 1	.357	.99
MAN 4740	Common Cathode-red	.400	.99	FND359	Common Cathode	.357	.75
MAN 4810	Common Anode-yellow	.400	.99	FND503	Common Cathode(FND500)	.500	.99
MAN 4840	Common Cathode-yellow	.400	.99	FND507	Common Anode (FND510)	.500	.99
MAN 6610	Common Anode-orange-D D	.560	.99	5082-7730	Common Anode-red	.300	1.30
MAN 6630	Common Anode-orange ± 1	.560	.99	HDSP-3400	Common Anode-red	.800	2.10
MAN 6640	Common Cathode-orange-D D	.560	.99	HDSP-3403	Common Cathode red	.800	2.10
MAN 6650	Common Cathode-orange ± 1	.560	.99	5082-7300	4 x 7 sgl. Digit-RHDP	.600	19.95
MAN 6660	Common Anode-orange	.560	.99	5082-7302	4 x 7 Sgl. Digit-LHDP	.600	19.95
MAN 6680	Common Cathode-orange	.560	.99	5082-7304	Overrange character (±1)	.600	15.00
MAN 6710	Common Anode-red-D.D.	.560	.99	5082-7340	4 x 7 Sgl. Digit-Hexadecimal	.600	22.50

RCA LINEAR

CA3013T	2.15	CA3082N	2.00
CA3023T	2.56	CA3083N	1.60
CA3035T	2.48	CA3086N	.85
CA3039T	1.35	CA3089N	3.75
CA3046N	1.30	CA3130T	1.39
CA3059N	3.25	CA3140T	1.25
CA3060N	3.25	CA3160T	1.25
CA3080T	.85	CA3401N	.49
CA3081N	2.00	CA3600N	3.50

CALCULATOR CHIPS/DRIVERS

MM5725	$2.95
MM5738	2.95
DM8864	2.00
DM8865	1.00
DM8887	.75
DM8889	.75
9374 7 seg. C.A. LED driver	1.50

CLOCK CHIPS

MM5309	$4.95
MM5311	4.95
MM5312	4.95
MM5314	4.95
MM5316	6.95
MM5318	9.95
MM5369	2.95
MM5387/1998A	4.95
MM5841	9.95

MOTOROLA

MC1408L7	$4.95
MC1408L8	5.75
MC1439L	2.95
MC3022P	2.95
MC3061P	3.50
MC4016(74416)	7.50
MC4024P	3.95
MC4040P	6.95
MC4044P	4.50

IC SOLDERTAIL — LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100
8 pin LP	$.17	.16	.15
14 pin LP	.20	.19	.18
16 pin LP	.22	.21	.20
18 pin LP	.29	.28	.27
20 pin LP	.34	.32	.30
22 pin LP	$.37	.36	.35
24 pin LP	.38	.37	.36
28 pin LP	.45	.44	.43
36 pin LP	.60	.59	.58
40 pin LP	.63	.62	.61

SOLDERTAIL STANDARD (TIN)

	1-24	25-49	50-100
14 pin ST	$.27	.25	.24
16 pin ST	.30	.27	.25
18 pin ST	.35	.32	.30
24 pin ST	.49	.45	.42
28 pin ST	$.99	.90	.81
36 pin ST	1.39	1.26	1.15
40 pin ST	1.59	1.45	1.30

SOLDERTAIL STANDARD (GOLD)

	1-24	25-49	50-100
8 pin SG	$.30	.27	.24
14 pin SG	.35	.32	.29
16 pin SG	.38	.35	.32
18 pin SG	.52	.47	.43
24 pin SG	$.70	.63	.57
28 pin SG	1.10	1.00	.90
36 pin SG	1.65	1.40	1.26
40 pin SG	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

	1-24	25-49	50-100
8 pin WW	$.39	.35	.31
10 pin WW	.45	.41	.37
14 pin WW	.39	.38	.37
16 pin WW	.43	.42	.41
18 pin WW	.75	.68	.62
22 pin WW	$.95	.85	.75
24 pin WW	1.05	.95	.85
28 pin WW	1.40	1.25	1.10
36 pin WW	1.59	1.45	1.30
40 pin WW	1.75	1.55	1.40

1/4 WATT RESISTOR ASSORTMENTS – 5%

ASST. 1	5 ea	10 OHM, 27 OHM	12 OHM, 33 OHM	15 OHM, 39 OHM	18 OHM, 47 OHM	22 OHM, 56 OHM	50 PCS	$1.75
ASST. 2	5 ea	68 OHM, 180 OHM	82 OHM, 220 OHM	100 OHM, 270 OHM	120 OHM, 330 OHM	150 OHM, 390 OHM	50 PCS	1.75
ASST. 3	5 ea	470 OHM, 1.2K	560 OHM, 1.5K	680 OHM, 1.8K	820 OHM, 2.2K	1K, 2.7K	50 PCS	1.75
ASST. 4	5 ea	3.3K, 8.2K	3.9K, 10K	4.7K, 12K	5.6K, 15K	6.8K, 18K	50 PCS	1.75
ASST. 5	5 ea	22K, 56K	27K, 68K	33K, 82K	39K, 100K	47K, 120K	50 PCS	1.75
ASST. 6	5 ea	150K, 390K	180K, 470K	220K, 560K	270K, 680K	330K, 820K	50 PCS	1.75
ASST. 7	5 ea	1M, 2.7M	1.2M, 3.3M	1.5M, 3.9M	1.8M, 4.7M	2.2M, 5.6M	50 PCS	1.75

ASST. 8R Includes Resistor Assortments 1-7 (350 PCS.) **$9.95 ea.**

$5.00 Minimum Order – U.S. Funds Only
California Residents – Add 6% Sales Tax
Spec Sheets – 25¢
1979 Catalog Available—Send 41¢ stamp

NEW 1979 Catalog

Jameco ELECTRONICS

PHONE ORDERS WELCOME (415) 592-8097

MAIL ORDER ELECTRONICS – WORLDWIDE
1021 HOWARD AVENUE, SAN CARLOS, CA 94070
Advertised Prices Good Thru December

TELEPHONE/KEYBOARD CHIPS

AY-5-9100	Push Button Telephone Dialler	$14.95
AY-5-9200	Repertory Dialler	14.95
AY-5-9500	CMOS Clock Generator	4.95
AY-5-2376	Keyboard Encoder (88 keys)	14.95
HD0165	Keyboard Encoder (16 keys)	7.95
74C922	Keyboard Encoder (16 keys)	9.95

ICM CHIPS

ICM7045	CMOS Precision Timer	24.95
ICM7205	CMOS LED Stopwatch/Timer	19.95
ICM7207	Oscillator Controller	7.50
ICM7208	Seven Decade Counter	19.95
ICM7209	Clock Generator	6.95

NMOS READ ONLY MEMORIES

MCM6571	128 X 9 X 7 ASCII Shifted with Greek	13.50
MCM6574	128 X 9 X 7 Math Symbol & Pictures	13.50
MCM6575	128 X 9 X 7 Alphanumeric Control Character Generator	13.50

MISCELLANEOUS

TL074CN	Quad Low Noise bi-fet Op Amp	2.49
TL494CN	Switching Regulator	4.49
TL496CP	Single Switching Regulator	1.75
11C90	Divide 10/11 Prescaler	19.95
95H90	Hi-Speed Divide 10/11 Prescaler	11.95
4N33	Photo-Darlington Opto-Isolator	3.95
MK50240	Top Octave Freq. Generator	17.50
DS0026CH	5Mhz 2-phase MOS clock driver	3.75
TIL308	.27" red num. display w/integ. logic chip	10.50
MM5320	TV Camera Sync. Generator	14.95
MM5330	4½ Digit DPM Logic Block	5.95
LD110/111	3½ Digit A/D Converter Set	25.00/set

LITRONIX ISO-LIT 1
Photo Transistor Opto-Isolator
(Same as MCT 2 or 4N25)
2/99¢

SN 76477
SOUND GENERATOR
Generates Complex Sounds
Low Power - Programmable
3.95 each

TV GAME CHIP AND CRYSTAL
AY-3-8500-1 and 2.01 MHZ Crystal (Chip & Crystal includes score display, 6 games and select angles, etc. **7.95/set**

EXAR

XR205	$3.40			XR2242CP	1.50
XR210	4.40			XR2264	4.25
XR215	4.40			XR2556	3.20
XR320	1.55			XR2567	2.99
XR-L555	1.50			XR3403	1.25
XR555	.39	XR1800	3.20	XR4136	1.25
XR556	.99	XR2206	4.40	XR4151	2.85
XR567CP	.99	XR2207	3.85	XR4194	1.45
XR567CT	1.25	XR2208	5.20	XR4202	3.60
XR1310P	1.30	XR2209	1.75	XR4212	2.05
XR1468CN	3.85	XR2211	5.25	XR4558	.75
XR1488	1.39	XR2212	4.35	XR4739	1.15
XR1489	1.39	XR2240	3.45	XR4741	1.47

DIODES

TYPE	VOLTS	W	PRICE	TYPE	VOLTS	W	PRICE
				1N4002	100 PIV	1 AMP	12/1.00
				1N4003	200 PIV	1 AMP	12/1.00
1N746	3.3	400m	4/1.00	1N4004	400 PIV	1 AMP	12/1.00
1N751	5.1	400m	4/1.00	1N4005	600 PIV	1 AMP	10/1.00
1N752	5.6	400m	4/1.00	1N4006	800 PIV	1 AMP	10/1.00
1N753	6.2	400m	4/1.00	1N4007	1000 PIV	1 AMP	10/1.00
1N754	6.8	400m	4/1.00	1N3600	50	200m	6/1.00
1N757	9.0	400m	4/1.00	1N4148	75	10m	15/1.00
1N759	12.0	400m	4/1.00	1N4154	35	10m	12/1.00
1N959	8.2	400m	4/1.00	1N4305	75	25m	15/1.00
1N965	15	400m	4/1.00	1N4734	5.6	1w	28
1N5232	5.6	500m	28	1N4735	6.2	1w	28
1N5234	6.2	500m	28	1N4736	6.8	1w	28
1N5235	6.8	500m	28	1N4738	8.2	1w	28
1N5236	7.5	500m	28	1N4742	12	1w	28
1N5242	12	500m	28	1N4744	15	1w	28
1N5245	15	500m	28	1N1183	50 PIV	35 AMP	1.60
1N456	25	40m	6/1.00	1N1184	100 PIV	35 AMP	1.70
1N458	150	7m	6/1.00	1N1185	150 PIV	35 AMP	1.70
1N485A	180	10m	5/1.00	1N1186	200 PIV	35 AMP	1.80
1N4001	50 PIV	1 AMP	12/1.00	1N1188	400 PIV	35 AMP	3.00

SCR AND FW BRIDGE RECTIFIERS

C36D	15A @ 400V	SCR(2N1849)	$1.95
C38M	35A @ 600V	SCR	1.95
2N2328	1.6A @ 300V	SCR	50
MDA 980-1	12A @ 50V	FW BRIDGE REC	1.95
MDA 980-3	12A @ 200V	FW BRIDGE REC	1.95

TRANSISTORS

PN2907 Plastic	7/1.00			2N3904	4/1.00
C106B1	.50	MJE3055	1.00	2N3905	4/1.00
MPSA05	.30	2N3392	5/1.00	2N3906	4/1.00
MPSA06	5/1.00	2N3398	5/1.00	2N4013	3/1.00
TIS97	6/1.00	PN3567	3/1.00	2N4123	6/1.00
TIS98	6/1.00	PN3568	4/1.00	PN4249	4/1.00
40409	1.75	PN3569	4/1.00	PN4250	4/1.00
40410	1.75	MPS3638A	5/1.00	2N4400	4/1.00
40673	1.75	MPS3702	5/1.00	2N4401	4/1.00
2N918	4/1.00	2N3704	5/1.00	2N4402	4/1.00
2N2219A	2/1.00	MPS3704	5/1.00	2N4403	4/1.00
2N2221A	4/1.00	2N3705	5/1.00	2N4409	5/1.00
2N2222A	5/1.00	MPS3705	5/1.00	2N5086	4/1.00
2N2369	5/1.00	2N3706	5/1.00	2N5087	4/1.00
2N2369A	4/1.00	MPS3706	5/1.00	2N5088	4/1.00
MPS2369	5/1.00	2N3707	5/1.00	2N5089	4/1.00
2N2484	4/1.00	2N3711	5/1.00	2N5129	5/1.00
2N2906	4/1.00	2N3724A	.65	PN5134	5/1.00
2N2907	5/1.00	2N3725A	1.00	PN5138	5/1.00
2N2925	5/1.00	2N3772	2.25	2N5139	5/1.00
MJE2955	1.25	2N3823	1.00	2N5210	5/1.00
2N3053	2/1.00	2N3903	5/1.00	2N5449	3/1.00
2N3055	.89	PN2222 Plastic	7/1.00	2N5951	3/1.00

CAPACITOR CORNER

50 VOLT CERAMIC DISC CAPACITORS

	1-9	10-99	100-		1-9	10-99	100-
10 pf	.05	.04	.03	.001µF	.05	.04	.035
22 pf	.05	.04	.03	.0047µF	.05	.04	.035
47 pf	.05	.04	.03	.01µF	.05	.04	.035
100 pf	.05	.04	.03	.022µF	.06	.05	.04
220 pf	.05	.04	.03	.047µF	.06	.05	.04
470 pf	.05	.04	.035	.1µF	.12	.09	.075

100 VOLT MYLAR FILM CAPACITORS

.001mf	.12	.10	.07	.022mf	.13	.11	.08
.0022	.12	.10	.07	.047mf	.21	.17	.13
.0047mf	.12	.10	.07	.1mf	.27	.23	.17
.01mf	.12	.10	.07	.22mf	.33	.27	.22

+20% DIPPED TANTALUMS (SOLID) CAPACITORS

.1/35V	.28	.23	.17	1.5/35V	.30	.26	.21
.15/35V	.28	.23	.17	2.2/25V	.31	.27	.22
.22/35V	.28	.23	.17	3.3/25V	.31	.27	.22
.33/35V	.28	.23	.17	4.7/25V	.32	.28	.23
.47/35V	.28	.23	.17	6.8/25V	.36	.31	.25
.68/35V	.28	.23	.17	10/25V	.40	.35	.29
1.0/35V	.28	.23	.17	15/25V	.63	.50	.40

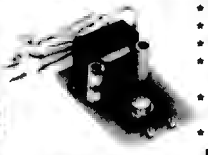
MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS

Axial Lead				Radial Lead			
.47/50V	.15	.13	.10	.47/25V	.15	.13	.10
1.0/50V	.16	.14	.11	.47/50V	.16	.14	.11
3.3/50V	.14	.12	.09	1.0/16V	.15	.13	.10
4.7/25V	.15	.13	.10	1.0/25V	.16	.14	.11
10/25V	.15	.13	.10	1.0/50V	.16	.14	.11
10/50V	.16	.14	.12	4.7/16V	.15	.13	.10
22/25V	.17	.15	.12	4.7/25V	.15	.13	.10
22/50V	.24	.20	.18	4.7/50V	.16	.14	.11
47/25V	.19	.17	.15	10/16V	.14	.12	.09
47/50V	.25	.21	.19	10/25V	.15	.13	.10
100/25V	.24	.20	.18	10/50V	.16	.14	.12
100/50V	.35	.30	.28	47/50V	.24	.21	.19
220/25V	.32	.28	.25	100/16V	.19	.15	.14
220/50V	.45	.41	.38	100/25V	.24	.20	.18
470/25V	.33	.29	.27	100/50V	.35	.30	.28
1000/16V	.55	.50	.45	220/16V	.23	.17	.16
2200/16V	.70	.62	.55	470/25V	.31	.28	.26

Circle 200 on inquiry card.

Jameco ELECTRONICS (New) Kits

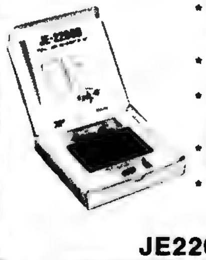
Regulated Power Supply



- Uses LM 309K
- Heat sink provided
- P.C. board construction
- Provides a solid 1 amp @ 5V
- Includes components, hardware and instructions
- Sizes: 3-1/2" x 5" x 2" high

JE200 $14.95

Function Generator Kit



- Provides 3 basic waveforms: sine, triangle & square wave
- Frequency range from 1 Hz to 100K Hz
- Output amplitude from 0-volts to over 6 volts (peak to peak)
- Uses a 12V supply or a ± 6V split supply
- Includes chip, P.C. board, components and instructions.

JE2206B $19.95

Digital Stopwatch Kit

- Use Intersil 7205 Chip
- Plated thru double-sided P.C. Board
- LED display (red)
- Times to 59 min. 59.59 sec. with auto reset
- Quartz crystal controlled
- Three stopwatches in one: single event, split (cummulative) and taylor (sequential timing)
- Uses 3 penlite batteries
- Size: 4.5" x 2.15" x .90"

JE900 $39.95

4-Digit Clock Kit

- Bright .357" ht. red display
- Sequential flashing colon
- 12 or 24 hour operation
- Extruded aluminum case (black)
- Pressure switches for hours, minutes and hold modes
- Includes all components, case and wall transformer
- Size: 3-1/4" x 1-3/4" x 1-1/4"

JE730 $14.95

6-Digit Clock Kit

- Bright .300 ht. common cathode display
- Uses MM5314 clock chip
- Switches for hours, minutes and hold functions
- Hours easily viewable to 20 feet
- Simulated walnut case
- 115 VAC operation
- 12 or 24 hour operation
- Includes all components, case and wall transformer
- Size: 6-3/4" x 3-1/8" x 1-3/4"

JE701 $19.95

Jumbo 6-Digit Clock Kit

- Four .630" ht. and two .300" ht. common anode displays
- Uses MM5314 clock chip
- Switches for hours, minutes and hold functions
- Hours easily viewable to 30 feet
- Simulated walnut case
- 115 VAC operation
- 12 or 24 hour operation
- Includes all components, case and wall transformer
- Size: 6-3/4" x 3-1/8" x 1-3/4"

JE747 $29.95

MICROPROCESSOR COMPONENTS

8080A/8080A SUPPORT DEVICES		
8080A	CPU	$ 9.95
8212	8-Bit Input/Output	3.25
8214	Priority Interrupt Control	5.95
8216	Bi-Directional Bus Driver	3.49
8224	Clock Generator/Driver	3.95
8226	Bus Driver	3.49
8228	System Controller/Bus Driver	5.95
8238	System Controller	5.95
8251	Prog Comm. 1/0 (USART)	7.95
8253	Prog Interval Timer	14.95
8255	Prog. Periph 1/0 (PPI)	9.95
8257	Prog DMA Control	19.95
8259	Prog Interrupt Control	19.95
6800/6800 SUPPORT DEVICES		
MC6800	MPU	$14.95
MC6802CP	MPU with Clock and Ram	24.95
MC6810API	128X8 Static Ram	5.95
MC6821	Periph Inter Adapt (MC6820)	7.49
MC6828	Priority Interrupt Controller	12.95
MC6830L8	1024X8 Bit ROM (MC68A30-8)	14.95
MC6850	Asynchronous Comm Adapter	7.95
MC6852	Synchronous Serial Data Adapt	9.95
MC6860	0-600 bps Digital MODEM	12.95
MC6862	2400 bps Modulator	14.95
MC6880A	Quad 3-State Bus Trans (MC8T26)	2.25
MICROPROCESSOR CHIPS—MISCELLANEOUS		
Z80(780C)	CPU	$19.95
Z80A(780-1)	CPU	24.95
CDP1802	CPU	19.95
2650	MPU	19.95
8035	8-Bit MPU w/clock, RAM, 1/0 lines	19.95
P8085	CPU	19.95
TMS9900JL	16-Bit MPU w/hardware, multiply & divide	49.95
SHIFT REGISTERS		
MM500H	Dual 25 Bit Dynamic	$.50
MM503H	Dual 50 Bit Dynamic	.50
MM504H	Dual 16 Bit Static	.50
MM506H	Dual 100 Bit Static	.50
MM510H	Dual 64 Bit Accumulator	.50
MM5016H	500/512 Bit Dynamic	.89
2504T	1024 Dynamic	3.95
2518	Hex 32 Bit Static	4.95
2522	Dual 132 Bit Static	2.95
2524	512 Static	.89
2525	1024 Dynamic	2.95
2527	Dual 256 Bit Static	2.95
2528	Dual 250 Static	4.00
2529	Dual 240 Bit Static	4.00
2532	Quad 80 Bit Static	2.95
2533	1024 Static	2.95
3341	Fifo	6.95
74LS670	4X4 Register File (TriState)	1.95
UART'S		
A-Y-5-1013	30K BAUD	5.95

MICROPROCESSOR MANUALS			
M-Z80	User Manual		$7.50
M-CDP1802	User Manual		7.50
M-2650	User Manual		5.00
ROM'S			
2513(2140)	Character Generator(upper case)		$9.95
2513(3021)	Character Generator(lower case)		9.95
2516	Character Generator		10.95
MM5230N	2048-Bit Read Only Memory		1.95
RAM'S			
1101	256X1	Static	$1.49
1103	1024X1	Dynamic	.99
2101(8101)	256X4	Static	3.95
2102	1024X1	Static	1.75
21L02	1024X1	Static	1.95
2111(8111)	256X4	Static	3.95
2112	256X4	Static MOS	4.95
2114	1024X4	Static 450ns	9.95
2114L	1024X4	Static 450ns low power	10.95
2114-3	1024X4	Static 300ns	10.95
2114L-3	1024X4	Static 300ns low power	11.95
5101	256X4	Static	7.95
5280/2107	4096X1	Dynamic	4.95
7489	16X4	Static	1.75
74S200	256X1	Static Tristate	4.95
93421	256X1	Static	2.95
UPD414 (MK4027)	4K	Dynamic 16 pin	4.95
UPD416 (MK4116)	16K	Dynamic 16 pin	14.95
TMS4044-45NL	4K	Static	14.95
TMS4045	1024X4	Static	14.95
2117	16,384X1	Dynamic 350ns (house marked)	9.95
MM5262	2KX1	Dynamic	4/1.00
PROM'S			
1702A	2048	FAMOS	$5.95
TMS2516 (2716)	16K*	EPROM(Intel 2716)	49.95
		*Requires single +5V power supply	
TMS2532	4KX8	EPROM	89.95
2708	8K	EPROM	10.95
2716 T.I	16K**	EPROM	29.95
		**Requires 3 voltages. —5V, +5V, +12V	
5203	2048	FAMOS	14.95
6301-1(7611)	1024	Tristate Bipolar	3.49
6330-1(7602)	256	Open C Bipolar	2.95
82S23	32X8	Open Collector	3.95
82S115	4096	Bipolar	19.95
82S123	32X8	Tristate	3.95
74186	512	TTL Open Collector	9.95
74188	256	TTL Open Collector	3.95
74S287	1024	Static	2.95

The Sinclair PDM35. A personal digital multimeter for only $59.95

Now everyone can afford to own a digital multimeter

A digital multimeter used to mean an expensive bulky piece of equipment.

The Sinclair PDM35 changes that. It's got all the functions and features you want in a digital multimeter yet they're neatly packaged in a rugged but light pocket-size case, ready to go anywhere.

The Sinclair PDM35 is supplied completely assembled with test leads and prods, protective wallet and Operator's Manual.

The Sinclair PDM35 is tailor-made for anyone who needs to make rapid measurements. Development engineers, field service engineers, lab technicians, computer specialists, radio and electronic hobbyists will find it ideal.

Technical specification

DC Volts (4 ranges)
Range: 1 mV to 1000 V
Accuracy of reading 1.0% ± 1 count
Note: 10 MΩ input impedance

AC Volts (40 Hz-5 kHz)
Range: 1 V to 500 V
Accuracy of reading 1.0% ± 2 counts

DC Current (6 ranges)
Range: 1 nA to 200 mA
Accuracy of reading 1.0% ± 1 count
Note: Max resolution 0.1 nA.

Resistance (5 ranges)
Range: 1Ω to 20 MΩ
Accuracy of reading 1.5% ± 1 count
Also provides 5 junction-test ranges.

Dimensions: 6 in x 3 in x 1½ in

Weight: 6 oz

Power supply: 9 V battery or Sinclair AC adapter (Battery not incl.)

Sockets: Standard 4 mm for resilient plugs

Options: AC adapter for 117 V 60 Hz power. De-luxe padded carrying wallet

XMAS SPECIAL - Get your PDM-35 PLUS the 117 volt AC Adapter and Padded carrying case for only $64.50 (Retail value $73.85)

B+K PRECISION 3½-Digit Portable DMM

- Overload Protected
- 3" high LED Display
- Battery or AC operation
- Auto Zeroing
- 1mv 1Va 0.1 ohm resolution
- Overange reading
- 10 meg input impendence
- DC Accuracy 1% typical

Ranges: DC Voltage - 0-1000V
AC Voltage 0-1000V
Freq Response 50-400 HZ
DC/AC Current: 0-100mA
Resistance 0-10 meg ohm
Size 6.4" x 4.4" x 2

Model 2800 $99.95

Comes with test leads, operating manual and spare fuse

Accessories:

AC Adapter BC-26	$9.00
Rechargeable Batteries BP-26	20.00
Carrying Case LC-26	7.50

CSC 100 MHz 8-Digit Counter

CONTINENTAL SPECIALTIES CORPORATION

- 20 Hz-100 MHz Range
- 6" LED Display
- Crystal-controlled timebase
- Fully Automatic
- Portable — completely self-contained
- Size — 1.75" x 7.38" x 5.63"
- Four power souces i.e batteries, 110 or 220V with charger, 12V with auto lighter adapter and external 7.2-10V power supply

MAX-100 $134.95

ACCESSORIES FOR MAX 100:

Mobile Charger Eliminator use power from car battery — Model 100 — CLA $3.95

Charger/Eliminator use 110 V AC — Model 100 — CAI $9.95

(New) CSC Mini-Max 6 Digit 50MHz Frequency Counter

- Guaranteed frequency range of 100 Hz to 50 MHz
- Full 6 digit display with antiglare window
- Fully automatic-range, polarity, slope, trigger, input level switching not required.
- Lead-zero blanking—All zeros to the left of the first non-zero digit are blanked. Kilo Hertz and Mega Hertz decimal points automatically light up when the unit is turned on.
- Built in input overvoltage protection.
- Use 9V Battery or 110/220V power.
- Complete with mini antenna.
- Lightweight — Only 8oz.

MINI-MAX $89.95

Accessories For Mini-Max

Part No.	Description	Price
MM-A4	Antenna	$ 3.95
MM-C5	Carrying case	5.95
MM-IPC	Input cable with clip leads	3.95
MM-AC2	110V adapter	9.95
MM-AC3	220V adapter	9.95

$5.00 Minimum Order – U.S. Funds Only
California Residents – Add 6% Sales Tax

Spec Sheets – 25¢
1979 Catalog Available—Send 41¢ stamp

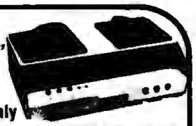
NEW 1979 Catalog

Jameco ELECTRONICS

PHONE ORDERS WELCOME (415) 592-8097

MAIL ORDER ELECTRONICS – WORLDWIDE
1021 HOWARD AVENUE, SAN CARLOS, CA 94070
Advertised Prices Good Thru December

The Incredible "Pennywhistle 103"



$139.95 Kit Only

The Pennywhistle 103 is capable of recording data to and from audio tape without critical speed requirements for the recorder and it is able to communicate directly with another modem and terminal for telephone "hamming" and communications. In addition, it is free of critical adjustments and is built with non-precision, readily available parts.

Data Transmission Method . . . Frequency-Shift Keying, full-duplex (half-duplex selectable)
Maximum Data Rate . . . 300 Baud
Data Format . . . Asynchronous Serial (return to mark level required between each character)
Receive Channel Frequencies . . . 2025 Hz for space, 2225 Hz for mark
Transmit Channel Frequencies . . . Switch selectable: Low (normal) = 1070 space, 1270 mark. High = 025 space, 2225 mark
Receive Sensitivity . . . 46 dbm accoustically coupled
Transmit Level . . . 15 dbm nominal. Adjustable from −6 dbm to 20 dbm
Receive Frequency Tolerance . . . Frequency reference automatically adjusts to allow for operation between 1800 Hz and 2400 Hz
Digital Data Interface . . . EIA RS-232C or 20 mA current loop (receiver is optoisolated and non-polar)
Power Requirements . . . 120 VAC, single phase, 10 Watts
Physical . . . All components mount on a single 5" by 9" printed circuit board. All components included.

Requires a VOM, Audio Oscillator, Frequency Counter and/or Oscilloscope to align

TRS-80 16K Conversion Kit

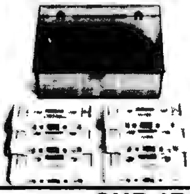
Expand your 4K TRS-80 System to 16K. Kit comes complete with:

* 8 each UPD416 (16K Dynamic Rams)
* Documentation for conversion

TRS-16K $115.00

Special Offer - Order both your TRS-16K and the Sup'R' MOD II Interface kit together (retail value $144.95) **for only $139.95**

COMPUTER CASSETTES

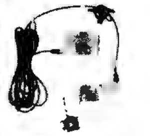


- 6 EACH 15 MINUTE HIGH QUALITY C-15 CASSETTES
- PLASTIC CASE INCLUDED 12 CASSETTE CAPACITY
- ADDITIONAL CASSETTES AVAILABLE #C-15-$2.50 ea

CAS-6 $14.95 (Case and 6 Cassettes)

SUP 'R' MOD II

UHF Channel 33 TV Interface Unit Kit



- Wide Band B/W or Color System
- Converts TV to Video Display for home computers, CCTV camera, Apple II, works with Cromeco Dazzler, SOL-20, IRS-80, Challenger, etc.
- MOD II is pretuned to Channel 33 (UHF).
- Includes coaxial cable and antenna transformer.

MOD II $29.95 Kit

Custom Cables & Jumpers

NEW NEW

DB 25 Series Cables

Part No.	Cable Length	Connectors	Price
DB25P-4-P	4 Ft.	2-DP25P	$15.95 ea.
DB25P-4-S	4 Ft.	1-DP25P/1-25S	$16.95 ea
DB25S-4-S	4 ft.	2-DP25S	$17.95 ea.

Dip Jumpers

DJ14-1	1 ft.	1-14 Pin	$1.59 ea.
DJ16-1	1 ft.	1-16 Pin	1.79 ea.
DJ24-1	1 ft.	1-24 Pin	2.79 ea.
DJ14-1-14	1 ft.	2-14 Pin	2.79 ea.
DJ16-1-16	1 ft.	2-16 Pin	3.19 ea
DJ24-1-16	1 ft.	2-24 Pin	4.95 ea.

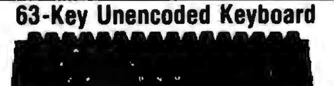
For Custom Cables & Jumpers, See JAMECO 1979 Catalog for Pricing

CONNECTORS

25 Pin-D Subminiature

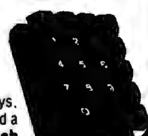
DB25P(as pictured)	PLUG	$2.95
DB25S	SOCKET	3.50
DB51226-1	Cable Cover for DB25 P or S	1.75

63-Key Unencoded Keyboard



This is a 63-key, terminal keyboard newly manufactured by a large computer manufacturer. It is unencoded with SPST keys, unattached to any kind of PC board. A very solid molded plastic 13 x 4" base suits most application. IN STOCK **$29.95/each**

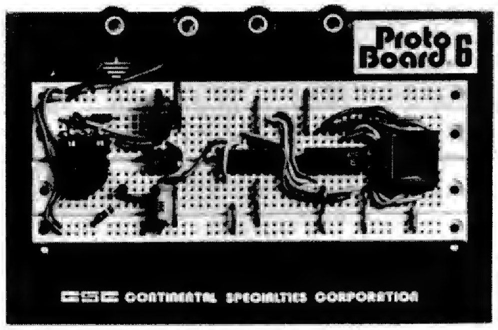
Hexadecimal Unencoded Keypad



19-key pad includes 1-10 keys, ABCDEF and 2 optional keys and a shift key. **$10.95/each**

Circle 200 on inquiry card.

Get Your Feet Wet with a Solderless Breadboard Without Wringing Your Wallet Dry



The PB-6 Pronto-Board Kit comes complete with a preassembled breadboarding socket, two preassembled solderless bus strips, four 5 way binding posts, a metal ground base plate, nonmarring feet and all required hardware. When complete, its 630 tie points permit flexible configurations of as many as six 14 pin dual-in-line package integrated circuits. Of the four binding posts, one is grounded to the ground base plate permitting high distributed capacitance and low distributed inductance for enhanced high speed circuit operation. The three remaining 5 way binding posts can be used to interconnect the circuit on the PB-6 to power and signal lines and the outside world. Kit is priced at $15.95 from Continental Specialties Corp, 70 Fulton Ter, New Haven CT 06509.■

Circle 545 on inquiry card.

Compucruise Reduces Fuel and Repair Expenses

Compucruise is an automotive microcomputer combining a 20 button backlighted keyboard, 5 digit blue fluorescent display, and appropriate sensors to provide a fuel management system, trip computer, clock and digitally displayed cruise control. Compucruise monitors speed, distance, fuel flow, time, battery voltage and choice of three temperatures, inside, outside or coolant. Its fuel management system indicates average fuel consumption, fuel used and remaining, plus distance and time to empty. The precision quartz crystal time computer features time, elapsed time, trip time, stopwatch and alarm. The trip computer displays distance, time and fuel to arrival. A total of 44 functions can be commanded by the touch of a button. Compucruise features cruise control that will accelerate your vehicle to any preselected road speed, can be instructed to adjust to traffic flow and has a resume feature. Either metric or English units can be displayed. With a command module no larger than a hand held calculator, Compucruise can be flush or bracket mounted, and complete hardware and installation instructions are included. The price is $189.95 from Zemco Inc, 1136 Saranap Av, Walnut Creek CA 94595.■

Circle 546 on inquiry card.

Master System Clock for LSI-11

A master system clock for use with Digital Equipment Corporation (DEC) LSI-11, LSI-11/2, and PDP-11/03 computer families has been announced by Nortek Inc, 2432 NW Johnson St, Portland OR 97210. The dual width module combines the features of a KW-11L real time clock, a KW-11P programmable clock, and adds an RT-11 compatible date and time clock.

An independently powered microprocessor helps insure that date, time and programmable count are maintained when the processor is not running. Simple operating system modifications eliminate the need for manually setting these values on power up.

13 programmable time rates from 1 MHz to once per hour are available. The programmable clock may also be used as an external event counter.

The basic unit with standard power supply and installation instructions is priced at $600. An optional battery backup power supply is available to provide protection against power failures for up to 24 hours.■

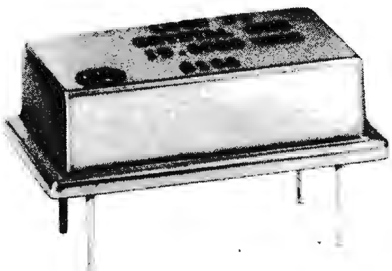
Circle 548 on inquiry card.

Low Cost Erasable Read Only Memory Eraser

The Information Central E-PROM Eraser is a 2 part unit consisting of a 2537 Å ultraviolet lamp and a base that holds up to two erasable read only memories. It operates from 115 VAC. The price is $45; Illinois residents should add 5% state tax. For more information contact Information Central Inc, 5521 N Broadway, Chicago IL 60640.■

Circle 547 on inquiry card.

Thick Film Crystal Clock Oscillator



LOCO II is a thick film crystal clock oscillator in dual-in-line (DIP) package form. Six discrete frequencies are available: 19.6608 MHz, 18.432 MHz, and 16.000 MHz.

This miniature oscillator is .820 by .520 by .250 inches (2.08 by 1.32 by .64 cm), operates from 5 VDC and drives 10 TTL gates.

The LOCO II provides master clock frequencies that can be divided to discrete universal asynchronous receiver transmitter (UART) frequencies for various data transmission rates, or they can be divided to provide multiple outputs which can drive combinations of transmission rate generators, microprocessors and LSI circuits in the same computer system.

The LOCO II is priced below $5. For further information, contact Motorola Component Products, 2553 N Edington, Franklin Park IL 60131.■

Circle 549 on inquiry card.

Attention Builders: Cases to House Prototype Electronic Circuits

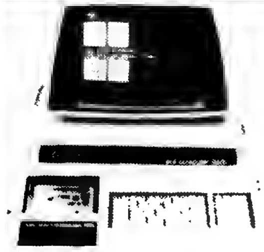
These Design Mate cases are designed to house prototype electronic circuits. They are made of high impact one-piece insulated plastic and feature a slope front panel, a metal bottom and include mounting screws.

There are two models of the Design Mate case: the DMC-1 measures 6.75 by 7.5 inches (16.15 by 19.05 cm) with a height that slopes from 1.5 to 3.25 inches (3.81 to 8.26 cm); the DMC-2 measures 5.63 by 6.0 inches (14.30 by 15.24 cm) with a height that slopes from 1.5 to 3 inches (3.81 to 7.62 cm).

The DMC-1 is $6.95 and the DMC-2 is $5.95. For further information contact Continental Specialties Corp, 70 Fulton Ter, New Haven CT 06509.■

Circle 550 on inquiry card.

BUY FROM THE BIGGEST



PET

IN STOCK NOW

$795

NCE/CompuMart

1250 North Main Street, Department KBC8
P.O Box 8610 Ann Arbor, Michigan 48107

THE PET CONNECTION

PET with 8K user memory	$795.00
Second Cassette drive	$99.95
RS-232 Printer Adapter — assembled board	$98.50
With power supply, case & DB25	$189.00
Beeper — PET controlled beep	$19.95
Video Buffer — add a conventional monitor	$19.95
BETSI — PET to S-100 I/O Motherboard Kit	$105.00
Assembled	$160.00

Hams! Add a PET to your rig.

Computerize your station with Microtronics M-65 Ham Interface and included software. You simply plug the unit into your PET and you're ready to display, transmit and receive either CW or RTTY at rates up to 100 WPM

ASSEMBLED KITS

$99.95

SECOND CASSETTE FOR PET $99

This peripheral from Commodore gives you the ability to use one tape deck for input and one for output copy files etc

SELECTRIC-BASED TERMINAL FOR PET

$895

We have interfaced our TC-71 to the PET Use with the IEEE-488 bus as a printer or with the VIA User Port for bidirectional I/O. Ideal for Word Processing applications

Sale price if bought with PET $795

LOW-COST PRINTER FOR PET

This is a completely refurbished Teletype KSR-33 terminal ready to attach to your PET's interface to use as a printer

INTRODUCTORY PRICE $395

READER/PUNCH

8 LEVEL PAPER TAPE/READER PUNCH

USED (90 day warranty) $175.00
Documentation Package $15.00

NEW ASCII INTERFACES for Reader/Punch

Boards & Cables	
Parallel	$220
RS232 Serial	$350
Board, Cables, Power Supply	
Parallel	$490
Serial	$615
Board, Cables, Power Supply	
Parallel	$575
Serial	$780

KIM-1

The original fully expandable single board computer

Assembled **$179**

with power supply **$209**

EXPAND YOUR KIM SYSTEM

KIMSI
KIM to S-100 Board

Kit	$125
Assembled	$169
Connector set	$15
KIM4 Expansion board	$119
MICRO TECH POWER SUPPLY Assembled	$34.95

KIM ENCLOSURE

Give your KIM a professional appearance. No alterations required: sturdy Kydex 100. Black

$23.50

WE WANT TO BUY USED MICROCOMPUTERS

Any Quantity

Urgently needed:

- Peripherals
- Manufacturers surplus
- Closed stores stock

CASH AVAILABLE

Call Today: (313) 994-3200

EXIDY SORCERER $895

- Z-80 Processor
- Full size ASCII keyboard
- User definable characters
- Up to 32K on-board RAM
- Interchangeable ROM PACs
- 30 x 64 display
- Resident 4K Monitor ROM
- Dual Cassette I/O
- Serial and Parallel I/O
- Complete Operators Manual
- Only $895

Top value in our catalog simply add a monitor and tape decks to complete the system. Can be used as a terminal. Later ROM PACs will offer Z-80 Assembly Language, APL, PILOT, Word Processing and a DOS PAC for disk based FORTRAN and COBOL. What a system for $895

[illegible]	$895
[illegible]	$1150
[illegible]	$1395
[illegible]	$299

VIDEO TERMINAL

SALE!

16 Ln x 64 chr display, 128 printable characters, software UART-(ASCII or Baudot) 3 serial interfaces - cassette I/O true composite video ASCII keyboard (parallel input) TTL compatible easy power connection

Assembled & tested **NOW $187**

Hazeltine 1400

Only **$749.00**

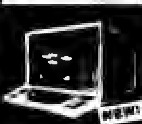
Now a 5th Generation in Terminals:

Introductory Sale Price — Immediate Delivery

- 24 x 80 display
- RS-232C interface
- 110 to 9600 Baud
- 12" CRT Monitor
- Remote screen control
- Uses 8048 microprocessor
- State-of-the-art design

A full capability terminal completely assembled with a 90 day warranty for less than you'd pay for a kit.

Compucolor II



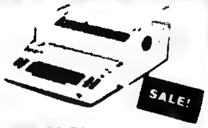
NEW!

Model 3 8K RAM $1495
Model 4 16K RAM $1795
Model 5 32K RAM $2395

- Beautiful 128 x 128 color graphics
- 4K Graphics RAM
- 16K ROM with Disk BASIC
- Built-in 51K mini floppy
- RS-232 Interface
- Full size keyboard
- Preprogrammed diskettes avail

TC-71 SELECTRIC TERMINALS

Top-quality printing terminal with RS-232 PTTC code. Includes desk, 130 char line. ASCII version employs 6802 for code conversion. See our catalog for details



ASCII	$874
Refurbished	$695
Working	$495
As-Is	$339
Acoustic Coupler	$159

SALE!

SALE ASCII INTERFACE FOR TC-71
30 day delivery Normally $199 NOW $179
Acoustic Coupler for use with ASCII TC-71 $159

SYM-1

- 28 Double function keypad including up to 24 Special functions
- Easy to view 6 digit hex led display
- KIM 1 hardware compatibility
- 6502 based system
- 4K byte ROM resident monitor and operating system
- Single 5 volt power
- 1K bytes of 2114 static RAM on sockets provided for immediate expansion to 4K bytes
- Enhanced software
- Standard interfaces include
  - Audio Cassette Recorder Interface with Remote Control
  - Full Duplex 20mA Teletype Interface
  - System Expansion Bus Interface
  - TV Controller Board Interface
  - CRT Compatible Interface
  - 50 I/O Lines in the basic system

SYM-1 From Synertek **$269.00**

SALE SANYO MONITOR

~~$285~~

VM4209 9" quality monitor $149

New Acoustic Coupler

$159

NCE Compumart Inc. breaks the price barrier with a new 300 baud RS-232C acoustic coupler. For only $159 you get an assembled, tested stand alone unit which accepts the standard telephone handset for full or half duplex operation. Complete with power supply and carrier detect light, the unit is housed in an attractive case for desk top use

TI PROGRAMMER

SALE!

Hexadecimal, octal, decimal calculator/converter

Reg. $59.95 NOW $53.95

PIXE-VERTER

$9.95

The original video to R.F. converter. Converts any good T.V. to a video monitor

HI PLOT™ DIGITAL PLOTTER

Surface area 8½ x 11 inches
Plot Area 7 x 10 inches
200 or 100 increments per inch. 240 steps per second at .01 inch or 480 steps per second at .005. EIA RS 232C (Serial) 8 Line TTL (Parallel)

$1,085

MATROX VIDEO RAMS S-100

24LN x 80CH	$295
256 x 256 dot	[illegible]

Autocomputer

Reduce fuel costs — monitor your engines' performance in real time

$129.50

The AUTOCOMP mounts in or on your dash, and by pushbutton control provides LED readouts of the following driving functions: Instantaneous MPG, Average MPF, Fuel Used, Distance Traveled, and Correct Time

*AUTOCOMP can be installed on most American and Japanese make autos and vans except those with fuel injection and is NOT recommended for European imports

Rockwell AIM65

- Fully assembled, tested and warranteed
- Expandable to 65K bytes (1K RAM on board)
- Alphanumeric

Options — 4K ROM Assembler Text Editor
8K Basic ROM

$375

10 DAY RETURN PRIVILEGE

- 90 day limited warranty
- Newest, most exciting products in personal computing
- Outstanding values
- All items fully assembled and tested unless otherwise stated.
- Send for our catalog for more detailed product information.

Just say charge it! Which card?

1-800 521 1534

- Michigan residents add 4% sales tax. • Foreign orders add 10% additional handling (U.S. funds only). • P.O.'s accepted from D&B rated companies. • Toll FREE phones open from 9:00 AM — 7:00 PM EST • Open accounts invited — call for credit application. • All orders add 4% shipping & handling. • Same day shipment on prepaid and credit card orders, except as noted. • Sorry — NO C.O.D.'s. • All prices subject to change without notice. • Write for giant FREE catalog.
- In the Ann Arbor area? Retail store open 11:00 AM — 7:00 PM Monday-Friday 10:00 AM — 5:00 PM Saturdays.

Circle 115 on inquiry card.

Electrolabs

PO Box 6721 Stanford, CA. 94305
415-321-5601
TLX 345567

✱ **TI Sockets 1 cent per pin.** All low profile solder tail 8 pin – 40 pin.

The "Pro" fully encoded ASCII KEYBOARD by Cherry. Auto REPEAT feature, 5 special function keys. 300mA/5V. *(Shown as mounted in 'The Case', Below)* **$119.00, 3/99.00, 10+/89.00**

USED SYLVANIA 12" MONITORS
Working: $69.95
Cold Chassis, 25lbs.
With P39 (Green) Tube 109

NEW P4 (White) Tube 99

The Dumb Terminal for Smart People

80X24 with full 128 char. ASCII UC+LC font with all control characters displayed. 300-19,200 baud RS232. 2nd font addressable from keyboard in you-program-it 2708 for APL, Graphics sets, etc. Plug in monitor I/O connector, 110VAC and you are ready. INCLUDES: 'The Case', Cherry Kbd. A used monitor, ESAT 200A, all options except vector addressable cursor and modem. Bulletproof design and construction. Normally $675.00 What you always wanted your ADM3 to be:
SYSTEM"A" $649.00 10/$599.00

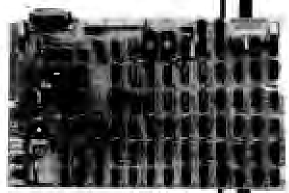
"The Case" Beautiful and sturdy anodized aluminum case in deep black designed to contain the ESAT 200A, and with a bezel cut out for the Cherry 'Pro' keyboard. (installed as shown above) Choose deep brown, light yellow, or crimson to accent or color code your installation. The only choice for hard-use institutional and educational applications. **$69.00, 10/ 59.00**

THE FANTASTIC!
MEMOREX FIVE-FIFTY

* *Hard and Soft Sectoring*
* *Single and Dual Density*
* *Double side configuration as a retrofit at any time.*
* *110/220V, 50/60Hz*
* *Pin for pin compatable with Shugart 800,801,850,851 (50 pin edge connector)*
* **PRICE BREAKTHROUGH! $425 EACH MERRY CHRISTMAS**

Double Sided Retrofit **$300**

LATEST REVISION: ESAT 200B (BILINGUAL)
Stand Alone Single Board Communicating Terminal
80X24 Standard Format
Split Speed 110 to 19,200 Baud
Two Fonts, 128 Char. ea. in 2708 EPROM
Full Feature Cursor, Page Xmit., Scrolling
5V@3A, 8"X14" Board. Requires only a keyboard and almost any monitor for the best terminal for the money anywhere!
Supplied with U&LC ASCII in one font.
$329.00



✱ *NEW!* **32 K, S-100 Universal Static Store.** Accepts 2114 RAMs or 70 ns, 3625 PROMs paging up to 8 Mby. Board only with manual and paging software $69.00. 32 Kby RAM 450 ns $679.00, 250 ns $789.00. We have software application notes for multi-task multi-user applications utilizing paging feature.

Shipping and Handling: Surface: $0.40/lb. **Air:** $0;75/lb., 1.00 minimum
Cal. Tax: 6.5% **Insurance:** $0.50 per $100.00

SSM

FORMERLY CYBERCOM/SOLID STATE MUSIC.

CB-1 8080 Processor Board. 2K of PROM 256 BYTE RAM power on/rest Vector Jump Parallel port with status. Kit 125.95
MB-6A Basic 8KX8 ram uses 2102 type rams, S-100 buss. Kit 450 NSEC ......$123.95. PCBD......$24.95
MB-7 16KX8, Static RAM uses μP410 Protection, fully buffered KIT........$299.95
MB-8A 2708 EROM Board, S-100, 8K8X or 16KX8 kit without PROMS $75.00
MB-9 4KX8 RAM/PROM Board uses 2112 RAMS or 82S129 PROM kit without RAMS or PROMS $72.00
IO-2 S-100 8 bit parallel /IO port, ⅔ of boards is for kludging. Kit ....$46.00 PCBD..........$25.95
IO-4 Two serial I/O ports with full handshaking 20/60 ma current loop: Two parallel I/O ports. Kit ....$130.00 PCBD $25.95
VB-1B 64 x 16 video board, upper lower case Greek, composite and parallel video with software, S-100. Kit..............$125.00 PCBD...............$25.95
Altair Compatible Mother Board, 11 x 11½ x ⅛". Board only ......$39.95. With 15 connectors......$94.95
Extended Board full size. Board only ....$ 8.95
With connector $12.95
SP-1 Synthesizer Board S-100
PCBD...............$39.95 KIT...............$135.95

82S23	$1.50	PRIME SUPPORT	
82S123	1.50	8080A	$ 9.95
82S126	1.95	8212	3.25
82S129	1.95	8214	6.50
82S130	3.95	8224	3.49
82S131	3.95	L2114 (450 NSEC)	7.25
MMI6330	1.50	L2114 (250 NSEC)	7.99
IM5600	1.50	2102AL-2	1.70
IM5603	1.95	2102AL-4	1.30
IM5604	3.95	21L02 (450)	1.25
IM5610	1.50	2708Q (National)..	8.95
IM5623	1.95	4116 (Apple Ram)	14.99
IM5624	3.95		

WMC inc. WAMECO INC.

FDC-1 FLOPPY CONTROLLER BOARD will drive shugart,pertek, remic 5" & 8" drives up to 8 drives, on board PROM with power boot up, will operate with CPM (not included).
DEC. & JAN. ONLY SPECIAL PCBD $23.95

FPB-1 Front Panel. (Finnally) IMSAI size hex displays. Byte or instruction single step. PCBD special, DEC. & JAN. $39.95

MEM-1 8KX8 fully buffered, S-100, uses 2102 type rams PCBD $24.95
QM-12 MOTHER BOARD, 13 slot, terminated, S-100 board only $34.95
CPU-1 8080A Processor board S-100 with 8 level vector interrupt PCBD $25.95
RTC-1 Realtime clock board. Two independent interrupts. Software programmable. PCBD $25.95
EPM-1 1702A 4K Eprom card PCBD $25.95
EPM-2 2708/2716 16K/32K EPROM CARD PCBD $24.95
QM-9 MOTHER BOARD, Short Version of QM-12. 9 Slots PCBD $30.95
MEM-2 16K x 8 Fully Buffered 2114 Board PCBD $25.95
16K RAM BOARD by HWE fully buffered, bank select standard to IEE buss gold fingers, solder mask, plated thru holes, silk screened ....PCBD $25.95
KLUDGE BOARD by HWE for S-100 glass epoxy over 2600 plated through holes, 4 regulators with CAPS all S-100 functions labeled, gold fingers. PCBD $29.95

MIKOS

419 Portofino Drive
San Carlos, California 94070
Please send for IC, Xistor and Computer parts list.

CHRISTMAS SALE (Dec. only)
5% OFF ON ALL KITS AND PCBD ON PREPAID ORDERS ONLY
(charge cards & C.O.D. are not included on this offer)

MIKOS PARTS ASSORTMENT WITH WAMECO AND CYBERCOM PCBDS

MEM-2 with MIKOS #7 16K ram with L2114 450 NSEC $249.95
MEM-2 with MIKOS #13 16K ram with L2114 250 NSEC $269.95

MEM-1 with MIKOS #1 450 NSEC 8K RAM $123.95
CPU-1 with MIKOS #2 8080A CPU 89.95
MEM-1 with MIKOS #3 250 NSEC 8K RAM 144.95
QM-12 with MIKOS #4 13 slot mother board 89.95
RTC-1 with MIKOS #5 real time clock 60.95
VB-1B with MIKOS #6 video board less molex connectors 99.95
EMP-1 with MIKOS #10 4K 1702 less EPROMS 49.95
EPM-2 with MIKOS #11 16-32K EPROMS less EPROMS 48.95
QM-9 with MIKOS #12 9 slot mother board 67.95

MIKOS PARTS ASSORTMENTS ARE ALL FACTORY PRIME PARTS. KITS INCLUDE ALL PARTS LISTED AS REQUIRED FOR THE COMPLETE KIT LESS PARTS LISTED. ALL SOCKETS INCLUDED.

VISA or MASTERCHARGE. Send account number, interbank number, expiration date and sign your order. Approx. postage will be added. Check or money order will be sent post paid in U.S. If you are not a regular customer, please use charge, cashier's check or postal money order. Otherwise there will be a two-week delay for checks to clear. Calif. residents add 6% tax. Money back 30 day guarantee. We cannot accept returned IC's that have been soldered to. Prices subject to change without notice. **$10 minimum order. $1.50 service charge on orders less than $10.00.**

Circle 229 on inquiry card.

NLS LEADS AGAIN!!

THE NEW MS-15 MINISCOPE

BATTERY OPERATED
15 MEGAHERTZ
OSCILLOSCOPE

FEATURES

- 15 megahertz bandwidth.
- External and internal trigger.
- Time Base — 0.1 microseconds to 0.5 Sec/div — 21 settings.
- Battery or line operation.
- Automatic and line sync modes.
- Power consumption less than 15W.
- Vertical Gain — 0.01 to 50 volts/div — 12 settings.
- Weight is only 3 pounds.

From the originator of the Digital Voltmeter, Non-Linear Systems comes the MS-15 Miniscope. It is a fine electronic instrument with a great deal of measuring capability and excellent accuracy. Its design is modern, utilizing the latest in low-powered integrated circuits, and it is packaged into the smallest practical size. The instrument fits into many briefcases and tool boxes with room to spare.

Operating characteristics have been chosen so that the MS-15 will make all of the measurements needed in servicing most electronic equipment. It is field-portable so its use is not restricted to the bench.

SPECIFICATIONS:

Y Axis

Vertical Input: x1-10mV/div to 10V/div in four ranges, each continuously variable. x2-20mV/div to 20V/div in four ranges, each continuously variable. x5-50mV/div to 50V/div in four ranges, each continuously variable. Accuracy is 3%

Input Impedance: 1M ohm shunted by 50 pF

Input Voltage: 350 volts peak

Bandwidth (3 db points): DC/DC to 15 MHz; AC 5 Hz to 15 MHz

Internal Calibrator: 1.0V (±5%) peak-to-peak square wave at 10-20 kHz

Graticule: 4 div Y by 5 div X; 1 div = 0.25 inch. Viewing area: 1.1"H x 1.35"W.

X Axis

Horizontal Input

Bandwidth (3 db points): DC to 200 kHz

Input Impedance: 1M ohm shunted by 50 pF

Sensitivity: 1V/div, 100V peak maximum input.

Time Base: x1, uS-0.1uS/div to 100 uS/div. x 2, uS-0.2uS/div to 200 uS/div. x5, uS-0.5uS/div to 500 uS/div. x1, mS-0.1mS/div to 100 mS/div. x2, mS-0.2mS/div to 200 mS/div. x5, mS-0.5mS/div to 500 mS/div. all in four ranges, each continuously variable. (Range increments are .1, 1, 10, 100.) With vernier in full clockwise position, calibrated time measurements are possible. Accuracy is 3%.

Triggering

Internal: Sweep triggered from internal trigger source.

Automatic: Trigger source is internal calibrator frequency. To be used if there is no other trigger source available to synchronize the sweep.

Line: Trigger is derived from line frequency when using the battery charger.

Slope: Selects sync to positive- or negative-going waveform.

Sensitivity: Less than one division.

External: Controls function as for internal triggering.

Power

Batteries: Three self-contained lead acid "D" cells.

Operating Time: Typically 4 hours.

Charging Time Scope Operating: Will run indefinitely but not reach full charge.

Non-operating: Sixteen hours.

External Power: Battery charger 115 vac (220 vac on request), 50-60Hz, less than 15 watts.

Dimensions: 3.1"H x 6.4"W x 8.0"D.

Weight: Three pounds.

Accessories Standard: Tilt stand, battery charger, input cable and miniature banana plugs (4).

Options: Leather carrying case and 10-1 attenuating probe, 10M ohm input.

Warranty: One year parts and labor.

MS-15 with Rechargeable Batteries and Charger

$318.00

Leather Carrying Case

The leather case has 2 separate compartments. One to hold the scope, the other to hold the charger, probe, shoulder strap, etc. The case can be worn on the belt, or over the neck.

The snaps used on the case are "one way", thus accidental striking of the case against an object will not undo the snaps or let it be pulled off your belt.

41-140 **$45.00**

Probe

10 to 1 probe with 10 megohm input. Probe uses spring hook tip for sure connection. Compensation network is located at the connector rather than at the probe, so as to keep size and weight to a minimum.

41-141 **$27.00**

FREE PROBE

Just send or mention this COUPON and **BYTE** MAGAZINE and receive 41-141 PROBE FREE with the purchase of MS-15 MINISCOPE.

TOLL FREE
1(800) 423-5633

PRIORITY ONE ELECTRONICS ©

16723B Roscoe Blvd. Sepulveda, CA 91343

Terms. VISA, MC, BAC, check, Money Order, C.O.D., U.S. Funds Only. CA residents add 6% sales tax. Minimum order $10.00. Orders less than $75.00 include 10% shipping and handling; excess refunded. Just in case . . . please include your phone no.

phone orders welcome (213) 893-8202

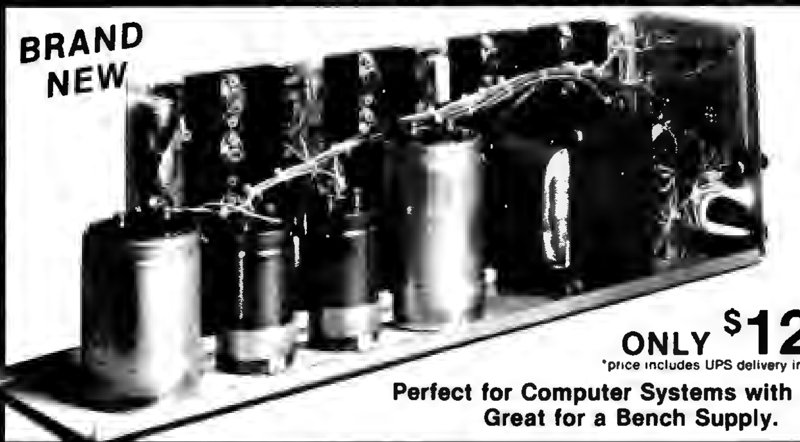
OEM and institutional inquiries invited.

Send for our latest brochure.

Visit our new retail location!

1 (800) 423-5633
CA., AK., HI., (213) 893-8202

BRAND NEW



COMPUTER POWER SUPPLY

Output Specifications
under continuous current loads

+ 5 Volts @ 17 Amps ± 2%, 50 mv ripple & noise maximum*

- 5 Volts @ 1.25 Amps ± 5%, 50 mv ripple & noise maximum
+ 12 Volts @ 3 Amps ± 3%, 150 mv ripple & noise maximum
12 Volts @ 2.75 Amps ± 3%, 150 mv ripple & noise maximum
+ 24 Volts @ 3.5 Amps ± 5%, 150 mv ripple & noise maximum

* + 5 Volt supply has remote sensing and overvoltage protection

Input

115VAC or 230 VAC, 48-63 Hz, complete with standard 115VAC 3 wire line cord.

ONLY $129.95*

*price includes UPS delivery in the continental US.

Perfect for Computer Systems with Discs.
Great for a Bench Supply.

We made a fantastic buy on these BRAND NEW Tele-Dynamic power supplies. They are shipped to you in the original factory cartons. Take advantage of this great offer and pay only a fraction of a large computer manufacturer's acquisition cost. Supply is limited, we suggest you phone your order in.

CONTINENTAL SPECIALTIES CORPORATION
CSC

Logic Probes and Digital Pulsers

LOGIC PROBES

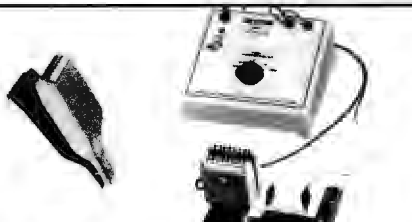
CSC logic probes are the ultimate tool for breadboard design and testing. These hand held units provide an instant overview of circuit conditions. Simple to use; just clip power leads to circuit's power supply, set logic family switch to TTL/DTL or CMOS/HTL. Touch probe to test node. Trace logic levels and pulses through digital circuits. Even stretch and latch for easy pulse detection. Instant recognition of high, low or invalid levels, open circuits and nodes. Simple, dual-level detector LEDs tell it quickly, correctly. HI (Logic "1"); LO (Logic "0"). Also incorporates blinking pulse detector, e.g., HI and LO LEDs blink on or off, tracking "1" or "0" states at square wave frequencies up to 1.5 MHz. Pulse LED blinks on for 1/3 second during pulse transition. Choice of three models to meet individual requirements: budget, project and speed of logic circuits.

MODEL LP-1
Hand-held logic probe provides instant reading of logic levels for TTL, DTL, HTL or CMOS. **Input Impedance:** 100,000 ohms. **Minimum Detectable Pulse:** 50 ns. **Maximum Input Signal (Frequency):** 10 MHz. **Pulse Detector (LED):** High speed train or single event. **Pulse Memory:** Pulse or level transition detected and stored.
CSC Model LP-1 Logic Probe – Net Each $44.95

MODEL LP-2
Economy version of Model LP-1. Safer than a voltmeter. More accurate than a scope. **Input Impedance:** 300,000 ohms. **Minimum Detectable Pulse:** 300 ns. **Maximum Input Signal (Frequency):** 1.5 MHz. **Pulse Detector (LED):** High speed train or single event. **Pulse Memory:** None.
CSC Model LP-2 Logic Probe – Net Each $24.95

MODEL LP-3
High speed logic probe. Captures pulses as short as 10 ns. **Input Impedance:** 500,000 ohms. **Minimum Detectable Pulse:** 10 ns. **Maximum Input Signal (Frequency):** 50 MHz. **Pulse Detector (LED):** High speed train or single event. **Pulse Memory:** Pulse or level transition detected and stored.
CSC Model LP-3 Logic Probe – Net Each $59.95

DIGITAL PULSER
The ultimate in speed and ease of operation. Simply connect clip leads to positive and negative power, then touch DP-1's probe to a circuit node; automatic polarity sensor detects circuit's high or low condition. Depress the pushbutton and trigger an opposite polarity pulse into the circuit. Fast troubleshooting includes injecting signals at key points in TTL, DTL, CMOS or other popular circuits. Test with single pulse or 100 pulses per second via built-in dual control push-button; button selects single shot or continuous modes. LED indicator monitors operating modes by flashing once for single pulse or continuously for a pulse train. Completely automatic, pencil size lab/field pulse generator for any family of digital circuits. **Output:** Tri-state. **Polarity:** Pulse-sensing auto-polarity. **Sync and Source:** 100 mA. **Pulse Train:** 100 pps. **LED Indicator:** Flashes for single pulse, stays lit for pulse train.
CSC Model DP-1 Digital Pulser – Net Each $74.95



LOGIC MONITOR 1
[illegible] open. Weight: [illegible]
CSC Model LM-1 Logic Monitor – Complete. $59.95

LOGIC MONITOR 2
Provides greater versatility and precision in testing all types of digital circuits. The fully isolated power supply and selectable trigger threshold let you match the precise characteristics of the logic family under test, permitting more accurate measurements. The connector/display unit clips over any DIP IC up to 16 pins. The power supply module contains the precision reference power supply and logic family selector switch. Operation is simple. Set the threshold switch to the proper logic family. Connect back clip lead to circuit ground. For CMOS circuitry, the red clip is connected to circuit positive or Vcc. The clip module is then clipped over the IC and the LED display instantly gives the logic states of all pins. **Logic Thresholds:** CMOS, 70% of circuit Vcc; HTL, 7.5 V; TTL, 2.4 V; DTL, 1.6 V; RTL, 1.2 V. **Maximum Visible Input Frequency:** 30 KHz at 50% duty cycle. **Size:** Test Clip, 4" l. x 2" w. x 1.75" d. when open; Power Module, 6" w. x 5.63" d. x 3" h. max. **Total Weight:** 20 ozs. **Power Required:** 117 VAC, 50/60 Hz, 10 W, also available for 220 VAC, 50, 60 Hz at slightly higher price.
CSC Model LM-2 Logic Monitor – Complete. $129.95

Digital IC Testers and COUNTER

MAX-100 PORTABLE FREQUENCY COUNTER
MAX 100 is a portable, high precision frequency counter that sets new standards in performance and value. In a compact, portable case, it gives you continuous readings from 20 Hz to a guaranteed 100 MHz, with 8-digit accuracy. Fast readings with 1/6-sec. update and 1 sec. sampling rate. Precise readings, derived from a crystal-controlled time base with 3 ppm accuracy. High sensitivity readings from signals as low as 30 mV, with diode overload protection up to 200 V peaks. Input signals over 100 MHz automatically flash the most significant digit, and to indicate low battery condition and extend battery life, the entire display flashes at 1 kHz.

SPECIFICATIONS

Frequency Range: 20 Hz to 100 MHz guaranteed; 110 MHz typical. **Gate Time:** 1 sec. **Resolution:** 1 Hz. **Accuracy:** ± 1 count + time base error. **Input Impedance:** 1 megohm shunted by 56 pF. **Coupling:** AC. **Sine Wave Sensitivity:** 30 mV RMS at 50 MHz. **Internal Time Base Frequency:** 3.579545 MHz crystal oscillator.
Stability: ± 3 ppm at 25° C. **Temperature Stability:** Better than 0.2 ppm °C, 0-50° C. Max. **Aging:** 10 ppm year. **Display:** Eight 0.6" LED digits. **Lead-Zero Blanking:** Decimal point appears between 6th and 7th digit when input exceeds 1 MHz. **Overflow:** With signals over 99,999,999 Hz, most significant (left hand) digit flashes, allowing readings in excess of 100 MHz. **Display Update:** 1/6-sec. plus 1 sec. gate time. **Low Battery Indicator:** When battery supply falls below 6.6 VDC, all digits flash at 1 Hz. **Power Required:** Internal, 6 "AA" cells; external, 110 or 220 VAC charger eliminator, auto cigarette lighter adapter or 7.2 10 VDC external supply. **Battery Charging:** 12-14 hrs. **Size:** 1.75" h. x 5.63" w. x 7.75" d. **Weight:** Less than 1.5 lbs. with batteries.
CSC Model MAX-100 Frequency Counter – Net Each $134.95

MAX-100 ACCESSORIES

Model 100-CLA – Mobile charger eliminator. Net Each	$ 3.95
Model 100-CA1 – 110 VAC charger eliminator. Net Each	9.95
Model 100-CA2 – 220 VAC charger eliminator. Net Each	9.95
Model 100-IPC – Input cable with clip leads. Net Each	5.95
Model 100-LLC – Low-loss tap-off	14.95
Model 100-MWA – Mini-whip antenna	3.95
Model 100-CC – Carry case	9.95

DIP JUMPERS

FLAT RIBBON CABLE ASSEMBLIES WITH DIP CONNECTORS

- **Available with 14, 16, 24 and 40 contacts.**
- **Mate with standard IC sockets.**
- **Fully assembled and tested.**
- **Integral molded-on strain relief.**
- **Line-by-line probeability.**

TOLL FREE 1(800) 423-5633

MATERIALS
DIELECTRIC: UL recognized glass-filled polyester.
CONTACTS: Non-corrosive copper alloy 770.
CABLE: All cable conductors are #28 AWG stranded 7/36 tin-coated copper with vinyl insulation. All cable is grooved top and bottom for easy tear down.

A P DIP Jumpers are the low-cost, high-quality solution for jumpering within a PC board; interconnecting between PC boards, backplanes and motherboards; interfacing Input/Output signals; and more.

All assemblies use rainbow cable. Standard lengths are 6, 12, 18, 24 and 36 inches.

DOUBLE-ENDED DIP JUMPERS

Arrow Denotes Pin No 1

No. Contacts	Length 6"	Length 12"	Length 18"	Length 24"	Length 36"
14	924106 6 R $2 41	924106-12 R $2 61	924106 18-R $2 82	924106-24-R $3.02	924106-36-R $3 43
16	924116 6 R $2 65	924116 12-R $2 88	924116-18-R $3 11	924116-24-R $3.34	924116 36-R $3.80
24	924126 6 R $4 15	924126-12 R $4 50	924126 18-R $4.85	924126-24-R $5 20	924126-36-R $5 90
40	924136 6-R $6 93	924136 12-R $7 52	924136-18-R $8 11	924136 24-R $8 70	924136-36-R $9 88

SINGLE-ENDED DIP JUMPERS

Arrow Denotes Pin No 1

No. Contacts	Length 36"
14	924102 36-R $2 33
16	924112 36 R $2 59
24	924122 36 R $4 00
40	924132 36 R $6 71

ACE • All-Circuit Evaluator for fast, solderless, plug-in circuit building and testing

Model 236

ACE Model No.	Tie Points	DIP Capacity	No. Buses	No. Posts	Board Size (inches)	Price Each
200-K (kit)	728	8 (16's)	2	2	4 9/16 x 5-9/16	[illegible]
208 (assem.)	872	8 (16's)	8	2	4-9/16 x 5-9/16	[illegible]
201-K (kit)	1032	12 (14's)	2	2	16 x 7	[illegible]
212 (assem.)	1224	12 (14's)	8	2	4 9/16 x 7	34 95
218 (assem.)	1760	18 (14's)	10	2	6 1/2 x 7-1/8	48 95
227 (assem.)	2712	27 (14's)	28	4	8 x 9-1/4	59 95
236 (assem.)	3648	36 (14's)	36	4	10-1/4 x 9-1/4	79 95

IC TEST CLIPS
TC-14 3 in. $4.50
TC-16 3 in. $4.75
TC-16 fits 14 & 16 Pin Dips

SALE

PRIORITY ONE ELECTRONICS ©

16723B Roscoe Blvd. Sepulveda, CA 91343

Terms VISA, MC, BAC check Money Order C.O.D. U.S. Funds Only. CA residents add 6% sales tax. Minimum order $10.00. Orders less than $75.00 include 10% shipping and handling, excess refunded. Just in case please include your phone no. Good thru 1978

We stock the entire A P Products "Faster and Easier" line

Send for our latest brochure

phone orders welcome (213) 893-8202

4116/416 16K Ram's 8 for $98.0
for APPLE, TRS-80, EXPANDO-RAM

Visit our new retail location!

HEATHKIT SOFTWARE DOCUMENTATION WRITER

At Heath Company quality documentation is an integral part of the final product. Since Heath is expanding in the computer products area, writers are needed to continue the Heath traditions in the software documentation areas.

We are seeking applicants with:

- An enthusiasm for personal computing
- A desire and ability to communicate software concepts
- Approximately four semesters of programming experience, including BASIC, one other high level language (preferably PASCAL), and the equivalent of one semester of assembly language
- A degree is preferred, but does not have to be in computer science

Heath offers a competitive salary with excellent fringe benefits, challenging opportunity and a pleasant work environment.

Send resume and salary requirements Or phone collect to:

H. T. Carson
HEATH COMPANY
Benton Harbor, Michigan 49022
(616) 982-3201

An Equal Opportunity Employer M/F

Circle 161 on inquiry card.

TRS-80 Z-80 8080

Professionally written software for the serious user.

Subroutines and interactive programs from $5 in the following areas:

Math	Science
Finance	Astronomy
Statistics	Electronics
Navigation	Investments
Real Estate	Amateur Radio
Music Analysis	Digital Design
Music Synthesis	Utility Functions

Performance Guaranteed.
Write for free catalog.

BENCHMARK
COMPUTING SERVICES
P. O. Box 385-B1
Providence, Utah 84332

Circle 31 on inquiry card.

TRS-80

16K MEMORY EXPANSION KIT............. $ 79.95

DISK DRIVES....... $ 399.00

CENTRONICS 779 TRACTOR PRINTERS.......$1179.00

TRS-80 & PET SOFTWARE.
SEND FOR FREE CATALOG.

APPARAT, INC.
6000 E. Evans Ave. Bldg. 2
Denver, CO 80222
303-758-7275

Circle 9 on inquiry card.

TRS-80 SOFTWARE

Moving Signboard	9.95
File Handling (16K RAM req.)	9.95
Loan Payment/Amortization	5.95
Prime Numbers	5.95
Buzzword	5.95
Day of the Week	5.95
Hamurabi	5.95
Pollution (16K RAM req.)	5.95
Tic-Tac-Toe	5.95
Slot Machine (with graphics)	5.95
Petals Around the Rose	5.95

20% off on order of three or more
One each of all of the above *49.95*

Business Systems

Advanced Amway Distributor Business System	119.95
Streamlined Amway Distributor Business System	49.95
Amway SA-1/SA-2R Verification Program	9.95

Circle Enterprises
P. O. Box 546
Groton, CT 06340

DEALER INQUIRIES WELCOMED!

Circle 48 on inquiry card.

RS232C Computer Compatible Paper Tape Transmitter/Model 612

Stops and starts on character at all speeds, uses manual control or X-on, X-off 90-260 volt, 50-60 Hz power. 50-9600 baud, up to 150 char/sec synchronous or asynchronous; gated internal or external clock, RS232C current loop or parallel output, reads 5-8 level tape, 7-11 frames per character, even or odd parity. Desk top or rack mount.

Addmaster Corporation
416 Junipero Serra Drive
San Gabriel, CA 91776
(213) 285-1121
Telex 674770 Addmaster SGAB

Circle 6 on inquiry card.

THE COMPUTER HARDWARE STORE INC.

Dealers for:
APPLE II, IMSAI
VECTOR GRAPHIC
KIM-I, OAE
CYBERNEX, JIM PAK

With
ALPHA-1 CASSETTE SYSTEM

16 Bits
ALPHA MICRO
TECHNICO STARTER SYSTEM

Catalog Available

818 FRANKLIN ST., ALEX., VA
703 548 8085

Circle 71 on inquiry card.

TRS-80 OWNERS

STATISTICAL PACKAGE ON CASSETTE – ONLY $8.95!! USER ORIENTED YET MORE SOPHISTICATED THAN STATISTICAL PROGRAMS COSTING 3X AS MUCH! FULL EXPLANATIONS AND TABLES INCLUDED.

–ANALYSIS OF VARIANCE– *3, 4, 5, or 6 groups!*
–CORRELATION– *(also provides N, means, & vars.)*
–T-TESTS– *3 types – each gives t, F, Ns, DFs Means, & Variances!*
–ANALYSIS OF COVARIANCE– *2 group design*
–MEAN, VARIANCE, STANDARD DEVIATION
EACH PROGRAM RUNS IN 4K!
– SPECIFY LEVEL I OR II

NEW!! THE FOLLOWING PROGRAMS REQUIRE 16K Level II:

DISASSEMBLER – *Shows machine code $15.*
BRIDGE GAME – *Play bridge! $15.*
GAME OF LIFE – *Dazzling displays $5.*

Programs & Systems developed on request.

M, M, & S SOFTWARE
16 MARILYN LANE
WESTBURY, NEW YORK 11590

NY State residents add 7% sales tax.

Circle 212 on inquiry card.

Get your "computer nut" the perfect gift. The original single board computer (an abacus) comes in a display case with "Break Glass in Emergency" printed on the front.

Send $14.95 plus $2.50 postage and handling to:

Intelligent Design
2535 Hayward Drive
Santa Clara CA 95051

Circle 181 on inquiry card.

New S-44E Bus Microprocessor Modules—Compatible with all Atwood and S-44 modules but feature 16 address lines. On-board address decoding. All bare boards $22.50.

- 8K RAM Board (2114s)
- Hi Resolute Video (6845)
- CPU Board (6502/6802)
- Color Video (6847)
- Real-Time Clock

S-44 Microprocessor Card Price List	free
S-44 Microprocessor Card Manual	$3.70
Ohio Scientific Challenger II, III Manual	9.00
PET Technical Manual (Our Own) with Schematics	11.00
KIM-1 User Manual	5.95
KIM Katalog (Our Own)	free
KIM Komputer Enclosure	59.95
PAL Computer Technical Manual	7.25
Technical Book Discount Price List	free
Computer Surplus Items Price List	free
Universal 16-bit Memory Brochure	free
2-Way Radio Price List	free
6500, 6800, and memory IC Price List	free
Repairs: KIM, SWTP, or PET Repaired for	52.50
Ohio Scientific Special	
New Challenger 1P with BASIC in ROM	329.95
Ohio Scientific Price List	free
Wanted: 6800 and 6502 Software	
Computer Class (Ohio Scientific)	Fri nite
Wind Generators and Solar Cells Catalog	2.00

SILVER SPUR
Electronic Communications Co
P. O. Box 385, Chino, CA 91710
or Visit Retail Store
13552 Central Avenue

- (714) 627-9386
- VISA
- Mastercharge
- Calif Residents Add 6% Sales Tax

Circle 327 on inquiry card.

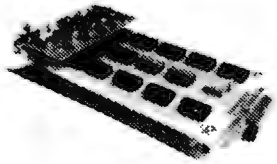
VISIBLE OR INFRA RED	USED FOR CHARACTER RECOGNITION FOR COMPUTERS WITH EXTERNAL CIRCUITS	MAY BE USED IN A VACUUM, UNDER WATER, HIGH ALTITUDE	IN MAGNETIC ENVIRONMENT BECAUSE THERE IS NO HIGH VOLTAGE OR MAGNETIC DEFLECTION

MINATURE SOLID STATE

202 VIDEO CAMERA KIT

FEATURING A. . 100 × 100 BIT SELF SCANNING CHARGED COUPLED DEVICE

THIS UNIQUE UPDATED CAMERA KIT FEATURES THE FAIRCHILD CCD 202C IMAGE SENSOR

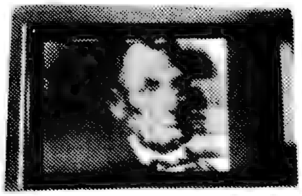


ADVANTAGES

- **IN THE FUTURE WE WILL SUPPLY A COMPUTER VIDEO INTERFACE CARD**
- All clock voltages operate at 6V reguiring no adjustments
- Higher video output signal
- We supply the power board, so only a 5V 1 Amp power source is needed
- The circuitry has been simplified for easier assembly
- Two level TTL output is supplied for interfacing

FEATURES

- Sensitive to infra red as well as visible light
- May be used for IR surveillance with an IR light source
- Excellent for standard surveillance work, because of light weight and small size
- All components mounted on parallel 3¾" × 6½" single sided boards
- Total weight under 1 lb.



We supply all semiconductors, boards, data sheets, diagrams, resistors and capacitors, and 8MM lens.
Sorry we do not supply the case, batteries and 5V supply.

$349.00 KIT
Add $75.00 to assemble and test
Add $2.00 Postage and Handling

UNIVERSAL 4Kx8 MEMORY BOARD KIT
$69.95
32-2102 1 fully buffered, 16 address lines, on board decoding for any 4 of 64 pages, standard 44 pin buss, may be used with F-8 & KIM

EXPANDABLE F8 CPU BOARD KIT
$99.00
featuring Fairbug PSU 1K-of static ram, RS 232 interface, documentation, 64 BYTE regis.er

C/MOS (DIODE CLAMPED)

4001	18	4019 –	37	4049 –	35	74C73	65
4002 –	18	4020 –	90	4050 –	35	74C74	45
4006 –	95	4021	90	4053 –	1 10	74C83	1 15
4007 –	18	4022 –	90	4055	1 25	74C86	40
4009	37	4023	18	4066 –	70	74C93	75
4010 –	37	4024 –	75	4071 –	18	74C151	1 40
4011 –	18	4025 –	18	4076 –	97	74C160	1 05
4012 –	18	4027 –	37	4520	70	74C161	1 05
4013 –	29	4028 –	80	74C00 –	22	74C174	1 05
4014	75	4029	95	74C02 –	22	74C175	1 05
4015 –	75	4030 –	33	74C04 –	24	74C193	1 20
4016 –	29	4035	97	74C08 –	22	74C901	48
4017	90	4042	65	74C10	27	74C902	48
4018	90	4046	1 35	74C42	85	74C914	1 70

1488 RS232 INTERFACE	$ 1 25	2114 4K STATIC RAM (450 ns)	[illegible]
1489 RS232 INTERFACE	$ 1 25	4116	$19 95
2509 TRI STATE STATIC SR	$ 1 35	[illegible] 4K DYN RAM	$ 3 40
2708 8K EPROM (450 ns)	$ 9 75	[illegible]	$ 3 95
2716 (TI)	$24 95	[illegible] 4K PROM	$ 4 95
2527 STATIC SHIFT REG	$ 1 95	[illegible]	$ 1 95
2513 CHARACTER GEN	$ 6 75	[illegible]	$ 3 25
2518 HEX 32 BIT SR	$ 7 20	AY 5 1013 UART	$ 5 75
2102 1 (450 ns)	$ 99	[illegible]	$ 4 95
21L02 1 (450 ns)	$ 1 25	[illegible] TELEDYNE 8 BIT A/D CONVERT	$ 9 95
[illegible] 4K RI DYN	$ 3 45	[illegible]	$ 2 75
MK 4008P	$ 1 95	[illegible]	$ 3 95
2101 1 256 x 4 STATIC	$ 2 45	[illegible]	$ 4 95
2111 1 256 x 4 STATIC	$ 3 45	[illegible]	$ 5 75
2112 1 256 x 4 STATIC	$ 2 75		

CRYSTALS $3.45 ea.
4 000 MHz
[illegible] 000 MHz
6 000 MHz
8 000 MHz
10 000 MHz

RIBBON CABLE FLAT (COLOR CODED) #30 WIRE
26 cond. – .50/per foot
40 cond. – .75 per foot
50 cond. – .90/per foot

CTS 206-8 eight position dip switch $1.60
CTS-206 4 four position dip switch . . $1.45
LIGHT ACTIVATED SCR's TO 18 200V 1A . . . $ 70

SILICON SOLAR CELLS
2¼" diameter .4V at 500 ma $4.00

LED READOUTS

FND 359 C.C. .4"	$ .50	DL-704 C.A. .3"	$ .75
FCS 8024 4 digit C C 8" display	$5 95	DL 747 C A .6"	$1 75
FND 503 C C .5"	$ 85	FND 803 C C .8"	$1 95
FND 510 C A .5"	$ 85	FND 810 C A .8"	$1.95
DL 704 .3 C C	$ 85		

PRINTED CIRCUIT BOARD
4 1 2" x 6 1 2" SINGLE SIDED EPOXY BOARD 1 16" thick, unetched $.60 ea. . . . . 5/$2.00

7 WATT LD-65 LASER DIODE IR $8.95

2N 3820 P FET	$ 45
2N 5457 N FET	$ 45
2N2646 UJT	$ 45
ER 900 TRIGGER DIODES	4 $1 00
2N 6028 PROG UJT	$ 65

MINIATURE MULTI-TURN TRIM POTS
100, 1K, 2K, 5K, 10K 20K, 50K, 100K, 200K, 500K, 1Meg, 2Meg, $ 75 each 3/$2.00

CHARGED COUPLE DEVICES
CCD 201C 100x100 Image Sensor $95.00
CCD 202C 100x100 Image Sensor $145.00
VERIPAX PC BOARD $4.00
This board is a 1/16" single sided paper epoxy board, 4½"x6½" DRILLED and ETCHED which will hold up to 21 single 14 pin IC's or 8,16 or LSI DIP IC's with busses for power supply connector.

FP 100 PHOTO TRANS . . . $ 50
RED, YELLOW, GREEN or AMBER LARGE LED's .2" . . . . 6/$1.00
TIL-118 OPTO-ISOLATOR . . . . $ .75
MOLEX PINS . . . . 100/$1.00, 1000/$8 00
1 WATT ZENERS 3 3, 4.7, 5.6, 9 1, 10, 12, 15, 18, or 22V . . . 6/$1.00
MC6860 MODEM CHIP $9 95
MCM 6571A 7 x 9 character gen ... $10.75

Silicon Power Rectifiers

PRV	1A	3A	12A	50A	125A	240A
100	06	14	30	80	3 70	5.00
200	07	20	35	1 15	4 25	6.50
400	09	25	50	1 40	6 50	9.50
600	11	30	70	1 80	8 50	12.50
800	15	35	90	2 30	10 50	16.50
1000	20	45	1 10	2 75	12 50	20 00

SAD 1024 a REDICON 1024 stage analog "Bucket Brigade" shift register. $18.95

IN 4148 (IN914) 15 $1 00

RS232 CONNECTORS
DB 25P male . . $2.25
DB 25S female . $2.95
HOODS $1 00

REGULATORS

309K	$ 95	340K 12, 15 or 24V	$ 95
723	$ 50	340T 5, 6, 8, 12 15, 18 or 24V	$ 95
LM 376	$ 60	78 MG	$1.35
320T-12, 15 or 24V	$1.25	79 MG	$1.35

TRANSISTOR SPECIALS

2N6233-NPN SWITCHING POWER	$ 1.95
MRF-8004 a CB RF Transistor NPN	$ .75
2N3772 NPN Si TO 3	$ 1.00
2N1546 PNP GE TO-3	$ .75
2N4908 PNP Si TO 3	$ 1 00
2N6056 NPN Si TO 3 Darlington	$ 1 70
2N5086 PNP Si TO 92	4 $ 1 00
2N3137 NPN Si RF	$ .55
2N3919 NPN Si TO 3 RF	$ 1 50
2N1420 NPN Si TO 5	3/$ 1.00
2N3767 NPN Si TO 66	$ 70
2N2222 NPN Si TO 18	5 $ 1 00
2N3055 NPN Si TO 3	$ .60
2N3904 NPN Si TO 92	6/$ 1.00
2N3906 PNP Si TO 92	6/$ 1.00
2N5296 NPN Si TO 220	$ 50
2N6109 PNP Si TO 220	$ 55
2N3638 PNP Si TO 5	5 $ 1 00
MPSA 13 NPN Si	4/$ 1.00

TTL IC SERIES

7400–	.13	7445–	65	74151–	61
7401–	13	7446–	68	74153–	61
7402–	13	7447–	.58	74154–	94
7403–	.13	7448–	68	74155–	58
7404–	15	7450–	15	74157–	55
7405–	13	7472–	25	74161–	55
7406–	16	7473–	28	74162	79
7407–	20	7474–	28	74163–	55
7408–	18	7475–	45	74164–	85
7409–	18	7476–	30	74165–	95
7410–	.13	7480–	31	74170–	1 68
7411–	18	7483–	65	74173–	1 20
7412–	13	7485–	87	74174–	95
7413–	36	7486–	28	74175–	85
7414–	60	7489	1 25	74176–	75
7416–	22	7490–	42	74177–	75
7417–	25	7491–	58	74180–	65
7420–	13	7492–	43	74181–	1 90
7425–	25	7493–	43	74190–	1 00
7426–	22	7494–	67	74191–	1 00
7427–	19	7495–	65	74192–	79
7430–	13	7496–	65	74193	79
7432–	22	74107–	28	74194–	80
7437	21	74121–	29	74195	50
7438	.21	74122–	38	74196–	86
7440	13	74123–	45	74279–	55
7441–	70	74125–	40	74367–	.50
7442–	37	74126–	40	75325	1 50
		74150–	94	75491–	50
				75492–	50

DATA CASSETTES 1/2 HR $ .95

22/44 Pin Solder Tail .156" Conn. $1.95

MM 5387AA new clock chip which will directly drive LED's 12/24 hrs., 1 supply & alarm $5 95

NO 30 WIRE WRAP WIRE SINGLE STRAND 100' $1.40

ALCO MINIATURE TOGGLE SWITCHES

MTA 106 SPDT	$ .95
MTA 206 DPDT	$ 1 70
MTA 206 P DPDT CENTER OFF	$ 1.85
MSD 206 P DPDT CENTER OFF LEVER SWITCH	$ 1.85

Full Wave Bridges

[illegible]

DIP SOCKETS

8 PIN	.17	24 PIN	.35
14 PIN	.20	28 PIN	.40
16 PIN	.22	40 PIN	.60
18 PIN	.25		

SANKEN AUDIO POWER AMPS

[illegible]

TANTULUM CAPACITORS

22UF	35V	5 $1 00	6 8UF	35V	4 $1 00
47UF	35V	5 $1 00	10UF	10V	$ 25
68UF	35V	5 $1 00	22UF	25V	$ 40
1UF	35V	5 $1 00	15UF	35V	3/$1.00
2 2UF	20V	5 $1 00	30UF	6V	5/$1 00
3 3UF	20V	4 $1 00	47UF	20V	$ 35
4 7UF	15V	5 $1 00	68 UF	15V	$ 50
			100 UF	10V	$ 40

74LS SERIES

[illegible]

LINEAR CIRCUITS

LM 101 –	75
LM 301 748	25
LM307 –	30
LM 308 –	75
LM 311 –	75
LM 318 –	1 20
LM 324 –	70
LM 339 –	1 10
LM 358 –	70
LM 370	1 15
LM 377 –	1 60
LM 380 –	95
LM 381 –	1 25
LM 382 –	1 25
LM 387	1 25
LM 537 –	2 50
LM 553 –	2 50
LM 555	39
LM 556 –	85
NE540L	2 25
560 –	2 00
565 –	95
566	1 10
567 –	1 10
703 –	90
[illegible]	[illegible]
[illegible]	[illegible]
709 –	25
710 –	35
741C or V –	25
747 –	50
LM 1310	2 50
1456 –	95
1458 –	50
3900 –	40
8038CC	3 90
791 –	1 95
LF356H –	1 20

TRIACS / SCR'S

[illegible]

Terms: FOB Cambridge, Mass
Send Check or Money Order.
Include Postage, Minimum Order $5.00, COD'S $20.00

Send 25¢ for our catalog featuring Transistors and Rectifiers
145 Hampshire St., Cambridge, Mass

SOLID STATE SALES
P.O. BOX 74B
SOMERVILLE, MASS 02143 TEL. (617) 547-7053

WE SHIP OVER 95% OF OUR ORDERS THE DAY WE RECEIVE THEM

Circle 340 on inquiry card.

Circle 395 on inquiry card.

Circle 76 on inquiry card.

Circle 346 on inquiry card.

Circle 23 on inquiry card.

Circle 377 on inquiry card.

Circle 74 on inquiry card.

Circle 284 on inquiry card.

Circle 373 on inquiry card.

Circle 148 on inquiry card.

P.O. Box 4430X Santa Clara, CA 95054
For will call only: (408) 988-1640
2322 Walsh Ave.

Quest ELECTRONICS

Same day shipment. First line parts only. Factory tested. Guaranteed money back. Quality IC's and other components at factory prices.

INTEGRATED CIRCUITS

7400 TTL	
7400N	.17
7402N	.17
7404N	.19
7409N	.23
7410N	.17
7414N	.63
7420N	.17
7422N	1.39
7430N	.20
7442N	.50
7445N	.69
7447N	.60
7448N	.69
7450N	.17
7474N	.29
7475N	.49
7485N	.88
7489N	2.00
7490N	.43
7492N	.43
7493N	.43
7495N	.69
74100N	.89
74107N	.29
74121N	.34
74123N	.59
74125N	.39
74145N	.60
74150N	.95
74151N	.69
74154N	1.00
74157N	.69
74161N	.87
74162N	.87
74163N	.87
74174N	.99
74175N	.90
74190N	1.15
74192N	.87
74193N	.85
74221N	1.55
74298N	1.65
74365N	.66
74366N	.66
74367N	.66

74LS00 TTL	
74LS00N	.25
74LS02N	.25
74LS04N	.25
74LS05N	.25
74LS08N	.25
74LS10N	.25
74LS13N	.40
74LS14N	.90
74LS20N	.25
74LS22N	.25
74LS28N	.41
74LS30N	.25
74LS33N	.39
74LS38N	.39
74LS74N	.35
74LS75N	.47
74LS90N	.51
74LS93N	.51
74LS95N	1.00
74LS107N	.35
74LS112N	.35
74LS113N	.35
74LS132N	.72
74LS136N	.35
74LS151N	.67
74LS155N	.67
74LS157N	.67
74LS162N	.91
74LS163N	.91
74LS174N	.95
74LS190N	1.06
74LS221N	1.95
74LS258N	.67
74LS367N	.80

LINEAR	
CA3045	.90
CA3046	.67
CA3081	1.80
CA3082	1.90
CA3089	2.95
LM301AN/AH	.35
LM305H	.87
LM307N	.35
LM308N	.89
LM309H	1.15
LM309K	.95
LM311H/N	.90
LM317T/K	2.92
LM318	1.35
LM320K-5	1.20
LM323K-5	6.95
LM320K-12	1.35
LM320K-15	1.35
LM320T-5	1.60
LM320T-8	1.60
LM320T-12	1.50
LM320T-15	1.60
LM324N	1.15
LM339N	1.55
LM340K-5	1.10
LM340K-8	1.10
LM340K-12	1.10
LM340K-15	1.10
LM340K-24	1.10
LM340T-5	1.10
LM340T-8	1.10
LM340T-12	1.10
LM340T-15	1.10
LM340T-18	1.10
LM340T-24	1.10
LM343H	4.50
LM370	1.15
LM377	4.50
LM379	5.00
LM380N	1.00
LM381	1.80
LM382	1.80
LM703H	.40
LM709H	.28
LM723H/N	.50
LM733N	.67
LM741CH	.35
LM741N	.25
LM747H/N	.62
LM748N	.35
LM1303N	.82
LM1304	1.10
LM1305	1.27
LM1307	2.00
LM1310	2.75
LM1458	.47
LM1800	1.75
LM1812	7.50
LM1889	3.00
LM2111	1.75
LM2902	1.50
LM3900N	.60
LM3905	1.75
LM3909N	.81
MC1458V	.50
NE540L	2.00
NE550N	.65
NE555V	.43
NE556A	.79
NE565A	1.00
NE566V	1.50
NE567V	1.20
NE570B	5.00
NE571B	5.00
78L05	.60
78L08	.60
79L05	.70
78M05	.85
75108	1.75
75491CN	.50
75492CN	.55
75494CN	.89

A to D CONVERTER	
8038B	4.50
8700CJ	13.95
8701CN	22.00
8750CJ	13.95
LD130	9.95
9400CJV/F	7.40
ICL7103	9.50
ICL7107	14.25

CMOS	
CD34001 Fair.	.50
CD4000	.16
CD4001	.21
CD4002	.21
CD4006	1.10
CD4007	.21
CD4008	.21
CD4009	.39
CD4010	.39
CD4011	.21
CD4012	.21
CD4013	.36
CD4014	.85
CD4015	.86
CD4016	.36
CD4017	.84
CD4018	.94
CD4019	.21
CD4020	1.02
CD4021	1.02
CD4022	.85
CD4023	.21
CD4024	.75
CD4025	.21
CD4026	1.51
CD4027	.38
CD4028	.79
CD4029	1.02
CD4030	.21
CD4035	1.02
CD4040	1.02
CD4042	.71
CD4043	.63
CD4044	.63
CD4046	1.67
CD4049	.36
CD4050	.36
CD4051	1.13
CD4060	1.42
CD4066	.71
CD4068	.40
CD4069	.40
CD4070	.40
CD4071	.21
CD4072	.21
CD4073	.21
CD4075	.21
CD4076	1.79
CD4078	.40
CD4081	.21
CD4082	.21
CD4116	.47
CD4490	5.50
CD4507	1.00
CD4508	4.25
CD4510	1.02
CD4511	.94
CD4515	2.52
CD4516	1.10
CD4518	1.02
CD4520	1.02
CD4527	1.51
CD4528	.79
CD4553	5.75
CD4566	2.25
CD4583	4.50
CD4585	1.10
CD40192	3.00
74C00	.28
74C04	.33
74C10	.28
74C14	2.10
74C20	.28
74C30	.28
74C48	1.95
74C74	.75
74C76	1.40
74C90	1.15
74C93	1.40
74C154	3.00
74C160	1.44
74C175	1.35
74C192	1.65
74C221	2.00
74C905	3.00
74C906	.75
74C914	1.95
74C922	5.50
74C923	5.50
74C925	6.95
74C926	6.95
74C927	6.95

INTERFACE	
8095	.65
8096	.65
8097	.65
8098	.65
8T09	1.25
8T10	4.80
8T13	3.00
8T20	5.50
8T23	3.10
8T24	3.50
8T25	3.20
8T26	1.69
8T28	2.75
8T97	1.69
8T98	1.69

MOS/MEMORY RAM	
2101-1	3.95
2102-1	.95
2102AL-4	1.60
21F02	1.25
2104A-4	4.95
2107B	4.95
2111-1	3.75
2112-2	3.95
2114	8.50
4116	17.95
25138	6.30
21L02-1	1.49
MM5262	.40
MM5280	3.00
MM5320	9.95
MM5330	5.94
PD411D-3	4.00
PD411D-4	5.00
P5101L	13.95
4200A	9.95
82S25	2.90
91L02A	1.75
HD0165-5	6.95
MM57100	4.50
GIAY38500-1	9.95
MCM6571A	9.95
9368	3.50
4100	10.00
416	16.00

CLOCKS	
MM5309	3.00
MM5311	3.00
MM5312	4.80
MM5313	3.80
MM5314	3.90
MM5315	4.00
MM5316	5.00
MM5318	3.90
MM5369	2.10
MM5841	14.45
MM5865	7.95
CT7001	5.80
CT7002	8.95
CT7010	8.95
CT7015	7.25
MM5375AA/N	3.90
MM5375AB/N	4.90
7205	16.50
7207	7.50
7208	15.95
7209	4.95
DS0026CN	3.75
DS0056CN	3.75
MM53104	2.50

MICROPROCESSOR	
6800	17.50
6802	24.95
8080A with data	8.95
8085	27.00
Z80	29.95
8212	2.90
8214	8.00
8216	2.90
8224	2.90
8228	5.35
8251	8.50
8253	10.00
8255	9.25
8257	19.50
8259	19.50
CDP1802CD	19.95
CDP1802D	25.00
CDP1861	12.95
6820	9.95
6850	12.95
6502	12.50

IC SOCKETS Solder Tin Low Profile

PIN	1 UP	PIN	1UP
8	.15	22	.30
14	.18	24	.35
16	.20	28	.42
18	.27	36	.58
20	.29	40	.57

3 level wire wrap gold
14 pin .25 16 pin .27
2 level 14 pin ww .20

WIRE WRAP LEVEL 3

PIN		PIN	
14	.25	24	.85
16	.27	28	1.00
18	.57	40	1.23

UART/FIFO	
AY5-1013	5.50
AY5-1014	7.50
3341	6.95

PROM	
1702A	3.95
N82S23	2.95
N82S123	3.50
N82S126	3.75
N82S129	3.75
N82S131	3.75
N82S136	8.75
N82S137	8.75
2708	12.50
DM8577	2.90
8223	2.90
2716T1	22.50
2716 Intel	48.00

CRYSTALS	
1 MHz	4.50
2 MHz	4.50
4 MHz	4.25
5 MHz	4.25
10 MHz	4.25
18 MHz	3.90
20 MHz	3.90
32 MHz	3.90
32768 Hz	4.00
1.8432 MHz	4.50
3.5795 MHz	1.20
2.0100 MHz	1.95
2.097152 MHz	4.50
2.4576 MHz	4.50
3.2768 MHz	4.50
5.0688 MHz	4.50
5.185 MHz	4.50
5.7143 MHz	4.50
6.5536 MHz	4.50
14.31818 MHz	4.25
18.432 MHz	4.50
22.1184 MHz	4.50

CONNECTORS	
44 pin edge	2.00
100 pin edge	4.50
100 pin edge WW	4.50

KEYBOARD ENCODERS	
AY5-2376	$12.50
AY5-3600	13.50
74C922	5.50
74C923	5.50
HD0165-5	6.95

IC Test Clips	14	16
Red	.85	.47
Black	.85	.47

Keyer 8043 14.50
comp. w/spec./mech.

TRANSISTORS	
2N1893	.40
2N2222A	.18
2N2369	.30
2N2904A	.20
2N2907A	.25
2N3053	.40
2N3638	.25
2N3643	.25
2N3904	.18
2N3906	.18
2N3055	.89
2N4400	.25
2N4401	.75
2N4402	.20
TIP31	.80
TIP33A	1.09

CLOCK MODULES Complete alarm clocks ready to hook up with transformer and switches. Very compact with .50" and .84" digits

MA1002A, C or E .50"	8.95
102P3 Transformer	2.25
MA1010A, C or E .84"	11.95
102P2 Transformer	2.25
Special transformer and six switches when purchased w/module	2.95
MA1003 car module .3" green fluor. display	16.95

RESISTORS ¼ watt 5%
10 per type .03 1000 per type .012
25 per type .025 350 piece pack
100 per type .015 5 per type 6.75

KEYBOARDS	
56 key ASCII keyboard kit	$62.50
Fully assembled	72.50
53 key ASCII keyboard kit	55.00
Fully assembled 65.00 Enclosure	14.95

LEDS	
Red T018	.15
Green, Orange, Yellow T018	.20
Jumbo Red	.20
Green, Orange, Yellow Jumbo	.25
Clipite LED Mounting Clips	8/$1.25

(specify red, amber, green, yellow, clear)

CONTINENTAL SPECIALTIES in stock
Complete line of breadboard test equip.
MAX-100 8 digit Freq. Ctr. $128.95

OK WIRE WRAP TOOLS in stock
Portable Multimeter $18.00

DIGITAL THERMOMETER $49.95
Batt. oper. General purpose or medical 32°–230°F. Disposable probe cover ±.2° accuracy. Comp. Assy. in compact case.

COMPUTER BOARD KITS	
8K RAM Board Kit	$134.95
4K EPROM Kit	114.95
I/O Board Kit	44.50
Extender Board w/connector	12.50
Video Interface board kit	125.00
16K EPROM board kit w/o PROMS	74.50
16K Static RAM board kit	395.00
North Star Floppy Disk Kit	$665.00
Additional Drive Kit	415.00

SPECIAL PRODUCTS	
MM5865 Stopwatch Timer	9.00
PC board	7.50
Switches Mom. Pushbutton	.27
3 pos. slide	.25
Encoder HD0165-5	6.95
3 Digit Universal Counter Board Kit Operates 5–18 Volt DC to 5 MHz typ. .125" LED display	10.50
Veiza selected switch	.80
Paratronics 100A Logic Analyzer Kit	$224.00
Model 10 Trigger Expander Kit	$229.00
Model 150 Bus Grabber Kit	$369.00

TRANSFORMERS	
6V 300 ma	3.25
12 Volt 300 ma transformer	1.25
12.6V CT 600 ma	3.75
12V 250 ma wall plug	2.95
12V CT 250 ma wall plug	3.50
24V CT 400 ma	3.95
10V 1.2 amp wall plug	4.85
18V 6 amp	12.95

DISPLAY LEDS			
MAN1	CA	270	2.90
MAN3	CC	125	.39
MAN72/74	CA/CA	300	1.00
DL704	CC	.300	1.25
DL707/DL707R	CA	300	1.00
DL727/728	CA/CC	500	1.90
DL747/750	CA/CC	600	1.95
DL750	CC	600	1.95
FND359	CC	357	.70
FND500/507	CC/CA	.500	1.35
FND503/510	CC/CA	500	.90
FND800/807	CC/CA	800	2.20
3 digit Bubble			.60
4 digit bubble			.80
DG8 Fluorescent			1.75
DG10 Fluorescent			1.75
5 digit 14 pin display			1.00
NSN69 9 digit display			.80
7520 Clairex photocells			.39
TIL311 Hex			9.50

COMPUTER GRADE CAPS			
1800	mfd	200V	4.50
2000	mfd	45V	2.50
3200		50V	2.50
5500		25V	2.50
5800		40V	3.00
6100		40V	3.00
6100		50V	3.00
7700		50V	2.50
8000		55V	2.50
9400		65V	3.00
10000		20V	2.50
12000		30V	3.50
35000		25V	5.00
55000		25V	5.00
82000		15V	6.00
160000		10V	6.00

QUEST Cosmac Super Elf Computer $106.95

Compare features before you decide to buy any other computer. There is no other computer on the market today that has all the desirable benefits of the **Super Elf** for so little money. The Super Elf is a small single board computer that does many **big** things. It is an excellent computer for training and for learning programming with its machine language and yet it is easily **expanded** with **additional memory, Tiny Basic, Ascii Keyboards, video character generation, etc.**

The **Super Elf** includes a **ROM** monitor for program loading, editing and execution with **SINGLE STEP for program debugging** which is not included in others at the same price. With **SINGLE STEP** you can see the microprocessor chip operating with the **unique Quest** address and data bus displays **before, during** and **after** executing instructions. Also, CPU mode and instruction cycle are shown on several LED indicator lamps.

An RCA 1861 **video graphics** chip allows you to connect to your own TV with an inexpensive video modulator to do graphics and games. There is a **speaker system** included for writing your own music or using many music programs already written. The speaker amplifier may also be used to drive relays for control purposes.

A 24 key **HEX keyboard** includes 16 HEX keys plus **load, reset, run, input, memory protect, memory select, monitor select** and **single step**. Large, on board displays provide output and optional **high and low address**. There is a 44 pin standard connector for PC cards and a 50 pin connector for the **Quest Super Expansion Board**. Power supply and sockets for **all** IC's are included in the price plus a detailed 90 page instruction manual.

Many schools and universities are using the Super Elf as a course of study. OEM's use it for training and research and development.

Remember, other computers only offer Super Elf features at additional cost or not at all. **Compare before you buy. Super Elf Kit $106.95, High address option $8.95, Low address option $9.95. Custom Hardwood Cabinet** with drilled and labelled front panel **$24.95. NiCad Battery Backup Kit $4.95.** All kits and options also come completely assembled and tested.

Questdata, a 12 page monthly software publication for 1802 computer users is available by subscription for $12.00 per year.

Tiny Basic for ANY 1802 System

Cassette $10.00. On ROM Monitor $38.00. Super Elf owners, 30% off. **Object code listing or paper tape with manual $5.50.**
Original ELF Kit Board $14.95.

Super Expansion Board with Cassette Interface $89.95

This is truly an astounding value! This board has been designed to allow **you** to decide how you want it optioned. **The Super Expansion Board comes with 4K of low power RAM** fully addressable anywhere in 64K with built-in memory protect and a **cassette interface**. Provisions have been made for all other options on the same board and it fits neatly into the hardwood cabinet alongside the **Super Elf**. The board includes slots for up to 6K of **EPROM** (2708, 2758, 2716 or TI 2716) and is **fully socketed** ($12.00 value). EPROM can be used for the monitor and Tiny Basic or other purposes.

A **1K Super ROM Monitor $19.95** is available as an on board option in 2708 EPROM which has been preprogrammed with a program loader/editor and error checking multi file cassette read/write software, (relocatable cassette file) another exclusive from Quest. It includes register save and readout, video graphics driver with blinking cursor and block move capability. The Super Monitor is written with subroutines allowing users to take advantage of monitor functions simply by calling them up. Improvements and revisions are **easily** done with the monitor. If you have the **Super Expansion Board** and **Super Monitor** the monitor is up and running at the push of a button.

Other on board options include **Parallel Input and Output Ports** with **full handshake**. They allow easy connection of an ASCII keyboard to the input port. **RS 232 and 20 ma Current Loop** for teletype or other device are on board and if you need more memory there are two **S-100** slots for static RAM or video boards. A Godbout 8K RAM board is available for $127.95. **Parallel I/O Ports $9.85, RS 232 $4.50, TTY 20 ma I/F $1.95, S-100 $4.50.** A **50 pin connector set** with ribbon cable is available at $12.50 for easy connection between the **Super Elf** and the **Super Expansion Board.**

The **Power Supply** for the Super Expansion Board is a 5 amp supply with +8v ± 18v + 12v – 5v. Regulated voltages are +5v & +12v $29.95. Deluxe version includes the case at $39.95.

Auto Clock Kit $15.95

DC clock with 4-.50" displays. Uses National MA-1012 module with alarm option. Includes light dimmer, crystal timebase PC boards. Fully regulated, comp. instructs. Add $3.95 for beautiful dark gray case. Best value anywhere

RCA Cosmac VIP Kit 229.00

Video computer with games and graphics.

Not a Cheap Clock Kit $14.95

Includes everything except case. 2-PC boards. 6-.50" LED Displays. 5314 clock chip, transformer, all components and full instrucs. Green and orange displays also avail. **Same kit w/.80" displays.** Red only. $21.95

60 Hz Crystal Time Base Kit $4.40

Converts digital clocks from AC line frequency to crystal time base. Outstanding accuracy. Kit includes: PC board, MM5369, crystal, resistors, capacitors and trimmer.

Digital Temperature Meter Kit

Indoor and outdoor. Switches back and forth. Beautiful. .50" LED readouts. Nothing like it available. Needs no additional parts for complete, full operation. Will measure –100° to +200°F, tenths of a degree, air or liquid. Very accurate. $39.95
Beautiful hardwood case w/bezel $11.75

NiCad Battery Fixer/Charger Kit

Opens shorted cells that won't hold a charge and then charges them up, all in one kit w/full parts and instructions. $7.25

PROM Eraser

Ultraviolet, assembled $49.95

Clock Calendar Kit $23.95

CT7015 direct drive chip displays date and time on .6" LEDS with AM-PM indicator. Alarm/doze feature includes buzzer. Complete with all parts, power supply and instructions, less case.

Sinclair 3½ Digit Multimeter

Batt./AC oper. 1mV and .1NA resolution. Resistance to 20 meg. 1% accuracy. Small, portable, completely assem. in case. 1 yr. guarantee. Best value ever! $59.95

Video Modulator Kit $8.95

Convert your TV set into a high quality monitor without affecting normal usage. Complete kit with full instructions.

2.5 MHz Frequency Counter Kit

Complete kit less case $37.50

30 MHz Frequency Counter Kit

Complete kit less case $47.75
Prescaler kit to 350 MHz $19.95

1978 IC Update Master Manual

1978 IC Update Master Manual **$24.50.** Complete IC data selector, 2175 pg. Master reference guide. Over 42,000 cross references. Free update service through 1978. Domestic postage $3.50. Foreign $6.00.

Stopwatch Kit $26.95

Full six digit battery operated. 2–5 volts. 3.2768 MHz crystal accuracy. Times to 59 min., 59 sec., 99 1/100 sec. Times std., split and Taylor. 7205 chip, all components minus case. Full instructions.

D Connectors RS232

25 Pin Subminiatures

DB25P	2.95	DE9P	1.50
DB25S	3.95	DE9S	1.95
Cover	1.50	DA15P	2.10
RS232 Complete Set	6.50	DA15S	3.10

S-100 Computer Boards

8K Static RAM Kit	$127.00
16K Static RAM Kit	265.00
24K Static RAM Kit	423.00
32K Dynamic RAM Kit	449.00
64K Dynamic RAM Kit	945.00
8K/16K Eprom Kit (less PROMS)	$89.00
Video Interface Kit	$139.00
Motherboard $39. Extender Board	$8.99

TERMS: $5.00 min. order U.S. Funds. Calif residents add 6% tax.
BankAmericard and Master Charge accepted.
Shipping charges will be added on charge cards.

FREE: Send for your copy of our NEW 1978 QUEST CATALOG. Include 28¢ stamp.

Compact Microcomputer Boards

A miniature printed circuit board for Motorola 6800 parts is now available. The single sided board measures 2 by 4 inches (5 by 10 cm) and has circuits etched for the 6802 processor, 6846 read only memory, 6810 programmable memory and 6850 ACIA IO port.

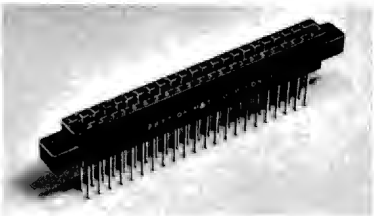
Other addresses (for additional programmable memory) may be enabled by painting the back of the card with photo resistant paint, and etching away the unused metal cladding. The board alone sells for $15.

Also available is a miniature, fully operational single board computer called the Ace, with 1 MHz clock, programmable timer, and a 2 K byte read only memory monitor. This full scale 6802 system provides for parallel data output rates of up to 50 k bps interleaved with serial output at rates of up to 500 k bps and comes completely assembled and tested with female connector for RS-232 and 20 pin IO power connector for $99.

For more information write Lumbert Computer Company, 1220 W Alameda #104, Tempe AZ 85282.■

Circle 576 on inquiry card.

Edge Connector Features Wire Wrapping Contacts



The Model Con-1 edge connector is a 44 pin connector for single or double sided circuit boards 1/16 inch (.16 cm) thick. The connector features .025 inch (.06 cm) square 3 level wire wrapping contacts on .156 inch (.40 cm) centers. The contacts are nickel silver over beryllium copper, and feature a bifurcated bellows design that provides constant pressure while minimizing contact distortion and stress. The connector body is molded of Underwriters' Laboratories and military approved Valox, an insulating material of dielectric, thermal and chemical characteristics. The Con-1 is priced at $3.49 and it can be obtained from OK Machine and Tool Corp, 3455 Conner St, Bronx NY 10475.■

Circle 575 on inquiry card.

Low Voltage Miniature Soldering Iron

The Soldercraft Model 6A is a miniature low voltage production soldering iron designed for versatile microcircuit and fine instrument work. This soldering iron, when powered by a multitap 18 W low voltage transformer, will provide controlled temperatures of 700° F at 6 V, 625° F at 5.5 V, 555° F at 5 V and 480° F at 4.5 V from its 3/32 inch (0.25 cm) tip. The heat is generated entirely within the tip, which provides maximum efficiency and faster heat recovery.

The Model 6A is priced at $6.90 and can be obtained by writing to the Mitchell-Hughes Co, 7534 Atoll Av, N Hollywood CA 91605.■

Circle 573 on inquiry card.

Automotive Computer Provides Driver Controlled Information Center

This self-contained driver-operated automotive computer instantly displays such data as *miles to go, vehicle location, estimated time of arrival, miles per gallon, cost per mile* and 19 other functions. The Prince On-Board Computer, which is 10½ by 2½ by 1¼ inches (26.67 by 6.35 by 3.18 cm) and weighs less than one pound (0.45 kg), is easily installed in cars, trucks or vans by connecting a speed transducer and fuel flow transducer, both supplied with the computer.

Function controls are color coded. The main programming keyboard and memory entry and recall bars are sized and located for easy access. Large 0.3 inch (0.76 cm) high intensity light emitting diode displays are recessed and filtered for optimum legibility day or night.

Other features of the unit include a memory scan and audio alarm which is automatically activated one mile before reaching a programmed location.

The unit costs $400. Contact the Prince Corp, POB 6, Holland MI 49423.■

Circle 574 on inquiry card.

Wire Wrap Jumpers Save Time

American Data Cable wire wrap jumper cables provide a means for making temporary or permanent electrical connections between 0.025 inch (0.0635 cm) square terminal posts, such as those commonly used in integrated circuit sockets and printed circuit connectors. Small components such as diodes, resistors, and capacitors can also be temporarily connected using the jumpers.

These cables offer an alternative to wire wrap or clip lead connections during the development and testing of electronic equipment. They are fully insulated and can be installed in a few seconds, generally without turning off equipment power.

These wire wrap jumpers are available in lengths from 4 inches (10.16 cm) to 6 feet (1.82 m) in five colors. Both assortments and color code by length bulk packs are available. Contact surfaces are gold, rated for 3 A continuous. The jumpers may be stacked up to three deep on a standard wire wrap pin. Insulated sleeving helps protect against shorts. Teflon wire and sleeving are available. Contact American Data Cable Inc, 903 San Antonio Rd, Los Altos CA 94022.■

Circle 572 on inquiry card.

CHRISTMAS GIFTS for YOUR FAVORITE COMPUTER

TWO NEW S100 PRODUCTS FROM WAMECO

FPB-1 FRONT PANEL BOARD

* DIRECT PLUG IN REPLACEMENT FOR IMSAI
* HEXADECIMAL READOUTS FOR DATA, ADDRESSES, AND PORT
* USES LOW COST SEVEN SEGMENT DISPLAYS
* CAN SINGLE STEP BY INSTRUCTION OR BY BYTE

$50 BARE

FDC-1 FLOPPY DISC CONTROLLER

* WILL CONTROL MINI OR FULL SIZE FLOPPY DISCS
* WILL CONTROL UP TO EIGHT FLOPPIES AS CONFIGURED, WITH SIMPLE MODIFICATION WILL CONTROL UP TO 64
* USES CPM

$45 BARE

OTHER WAMECO INC. S-100 PRODUCTS (bare board price)

* QMB-9 Nine Slot Mother Board $35.
* QMB-12 Thirteen Slot Mother Board $40.
* MEM-1A 8 KBYTE 2102 Ram Memory Board $30.
* MEM-2 16 KBYTE 2114 Ram Memory Board $30.
* EPM-1 4 KBYTE 1702 EPROM Board $30.
* EPM-2 16 or 32 KBYTE 2708/2716 EPROM Board $30.
* RTC-1 RealTime Clock Board $30.
* CPU-1 8080 CPU Board $30.

WAMECO DEALERS AND DISTRIBUTORS:

AUSTRIA
Margreiter Data System
A-1070 Wien GMBH
Schottenfeld Gasse 11
Phone (0222) 96 15 65

CANADA
Stephen Fielding
Hamilton Logic Systems
62 Robins Avenue
Hamilton, Ontario L8H 4N4

F. Jackson Electronics Co.
Box 70, Site 2, RR#3
Aamdale, Halifax County
Nova Scotia B3L 4J3
Phone 876-2920

* Orthon Computers
12411 Stony Plain Road
Edmonton, Alberta T5N 3N3
Phone (403) 488-2921

Robo-Tronics
509 16 Avenue NW
Calgary T2VROJ6
Phone (403) 282-9468

The Computer Shop
3515 18th Street S. V.
Calgary T2T 4T9

DENMARK
Piezodan Aps.
Bakkedraget 55
3480 Fredensborg
Phone 03-283744

ENGLAND
Leenshire Ltd.
13 Cathedral View
Winchester SO23 8PR

FRANCE
Euro Computer Shop
16, Rue Louis Pasteur
92100 Boulogne
Paris
Phone Paris 825-82-52

GERMANY
ABC Computer-Shop GMBH
Schellingstr. 33
8 000 Munchen 40
Phone 089/282892

JAPAN
Onward Enterprise Corp.
Suehirorocho Heim Bldg. Rm 706
5-5, 3-Chome, Sokokanda,
Chiyoda-Ku Tokyo, 101
Phone (03) 255-9885

SCOTLAND
AIRAMCO Ltd.
30, Witches Linn, Ardrossan
Ayrshire, KA22-8BR
Phone 0294-65530

UNITED STATES

ARIZONA
* Computer Ware, Inc.
214 West Southern
Tempe, 85282
Phone (602) 968-6312

CALIFORNIA
Adares Micro Computer Systems
239 Frances Street
Bakersfield, 93308
Phone (805) 393-1326

* Advanced Computer Products
1310 "B" East Edinger
Santa Ana, 92705
Phone (714) 538-8813

Anchor Electronics
2102 Walsh Ave.
Santa Clara
Phone (408) 246-3696

Byte Shop Computer Store
3400 El Camino Real
Santa Clara
Phone (408) 249-4221

Byte Shop of Palo Alto
2233 El Camino Real
Palo Alto, 94306
Phone (415) 327-8080

Computrend
Box 2541-K
Anaheim, 92804
Phone (714) 533-3572

D.C.I. Computer Systems
4670 N. El Capitan
Fresno
Phone (209) 266-9566

* Electronics Systems
P.O. Box 9641
San Jose, 95157
Phone (408) 226-4064

Hobby World
5716 West Manchester #5
Los Angeles, 90045
Phone (213) 641-4200

Hollywood Systems
9100 Sunset Blvd.
Suite 112
Hollywood, 90069
Phone (213) 271-9726

* Jade Computer Products
4901 West Rosecrans Blvd.
Hawthorne, 90250
Phone (213) 679-3313

I.C.E. House, Inc.
P.O. Box 336
San Bernardino
Phone (714) 888-3690

* Mikos
419 Portofino Avenue
San Carlos, 94070

PCE Electronics
4782 Dewey Drive
Fair Oaks, 95628
Phone (916) 966-7033

Proko Electronics
San Luis Obisbo, 93401
Phone (805) 544-5441

COLORADO
Bits and Chips
718 9th Street
Greeley
Phone (303) 356-7554

Byte Shop of Colorado Springs
1845 N. Circle Drive
Colorado Springs, 80909
Phone (303) 633-7075

The Computer Broker
12805 W. Stanford
Morrison, 80465
Phone (303)979-6441

Computer Technology
3400 West 64th Ave.
Denver, 80221

Miti Mini Computer Co.
621 S. Broadway
Denver, 80209
Phones (303) 778-6230,
733-8681

HAWAII
Mahalo Microsystems Limited
355 Royal Hawaiian Ave.
Honolulu
Phone (808) 922-2152

NEW JERSEY
William Electronics Supply
1863 Woodbridge Avenue
Edison, 08817
Phone (201) 985-3700

MINNESOTA
P.S. Inc.
1625 First Avenue North
Moorhead, 56560
Phone (218) 233-6682

PENNSYLVANIA
Marketline Systems
2337 Philmont Ave.
Huntington Valley
Phone (215) 947-6670

Micro-Computer Products
116 S. Pugh Street
State College, 16801
Phone (814) 238-7711

TEXAS
Young Electronics Service
P.O. Box DD
College Station, 77840
Phone (713) 693-3462

WEST VIRGINIA
The Computer Corner
22 Beechurst Ave.
Morgantown, 26505
Phone (304) 292-9700

* DISTRIBUTOR

Dealer Inquiries Invited !
University Discounts Available

WMC inc. WAMECO Inc. 3107 LANEVIEW DRIVE, SAN JOSE, CALIF. 95132

Circle 388 on inquiry card.

RONDURE COMPANY

2522 BUTLER ST.
DALLAS, TEXAS 75235
214–630-4621

the computer room

ASCII SELECTRIC

SPECIAL SALE
$875.00

Used Working and Clean

TESTED WITH NEW ASCII ELECTRONICS
Factory Reconditioned
$1075.00

ASCII Selectric with ASCII parallel electronics. Immediate Delivery—Shipped from inventory.

Printer Mechanism: Heavy duty input/output, Series 745.

Weight: 120 lbs. Dimensions: 29"Hx35"Wx33"D.

Print Speed: (14.8 characters per second)

Platen: 15" wide, pin feed or form feed device optional (132 print positions).

Parallel output only—15 characters per second accepts 7 bit ASCII parallel w/strobe & prints on Selectric. The unit still works as a typewriter in off-line mode.

DATEL SELECTRIC (IBM Selectric Mechanism)

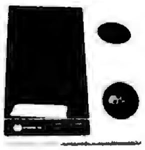
"As Is" Complete	$395.00
Working	$495.00
Reconditioned	$695.00

Specifications:
- Size: 21"Wx21"Dx8"H.
- Power Input 115 Volt Hz
- Interface: RS232
- Weight: 54 lbs. (Shipping weight 65 lbs.)
- 15" Carriage
- 15 CPS
- Correspondence code
- Half Duplex
- 132 Print Positions, 10 Pitch

Used – $395

NOVATION DC3102A

Used Working



$150.00

RS232 Connection
300 Baud.

TI 990/4

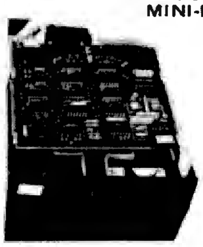
Single Board 16 Bit Micro Computer

NEW $250.00

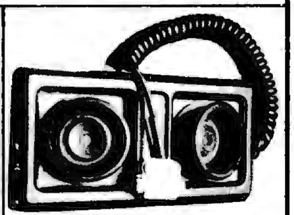
USED MODEMS & COUPLERS

NOVATION TC102 (Acoustic)	$45.00
AJ 233 (Acoustic)	$25.00
CF 318 (Hard Wire)	$25.00

SHUGART MINI-FLOPPY DRIVE



NEW PRICE
$325.00 ea.
Model SA-400



FLAT PACK ACOUSTICAL MODEM PICK-UP

Useable with most modem chips/kits
Used — $17.50 (w/prints)

ORDERING INFORMATION:
We ship the same day we receive a certified check or money order.
Texas residents add 5% sales tax. Please call if you have a question.
Write for our CATALOG of many parts, terminals, printers, etc.
All items subject to availability. Your money returned if we are out of stock.

SHIPPING INFORMATION:
Modems: $2.00 each; 2 for $4.00 UPS.
Large Items & Parts: Specify Freight or Air Freight Collect
Foreign Orders: Add appropriate freight or postage.
We now take Master Charge and Visa orders, Specify full number, bank number and expiration date.

ATTENTION TRS-80 & APPLE USERS

IN TIME FOR CHRISTMAS

A PRINTER FOR YOUR COMPUTER

TERMS: VISA, MASTERCHARGE,
Cashier Check or Money Order.
C.O.D. with 10% down.
Shipping Via Air or Truck collect.

3 S SALES
P.O. BOX 45944
TULSA, OK 74145
918/622-1058

$995.00
MODEL 3S-80 for TRS-80
Ready to plug into your expansion interface.

MODEL 3S-PP
for computers with 8 bit serial port.

MODEL 3S-SS
for computers with RS-232 port.

$1095.00 for
MODEL 3S-AA
Includes RS-232 card for AppleII
Specify model number on order.

- Ready to plug into your computer
- Very high quality print
- Completely refurbished IBM 731 I/O Selectric terminal in a new table
- Upper & lower case removable type ball
- Special I/O interface
- Heavy duty re-mfg. IBM power supply

BECKIAN ENTERPRISES

All Prime Quality — New Parts Only

Satisfaction Guaranteed

EDGE CARD CONNECTORS: GOLD PLATED.
BODY: Non brittle, solvent resistant, high temp, G.E. Valox. The finest you can buy.
CONTACTS: Bifurcated Phos./Bronze; Gold/Nickel.

ALTAIR S-100: Cont./Ctrs. .125" Row Spacing, .140"

50/100 Dip Sold.	$3.95 ea.	5 pcs.	$3.75 ea.
50/100 Sold. Eye	6.95 ea.	5 pcs.	6.50 ea.

IMSAI S-100: Cont./Ctrs. .125" Row Spacing, .250"

50/100 Dip Sold.	$4.20 ea.	5 pcs.	$3.95 ea.
50/100 W/Wrap 3	3.75 ea.	5 pcs.	3.50 ea.
IMSAI CARD GUIDES:	0.19 ea.	5 pcs.	0.16 ea.

CROMEMCO S-100: Cont./Ctrs. .125" Row Spacing, .250"

50/100 Dip Sold. (Or short W/Wrap)	$6.50 ea.	5 pcs.	$6.00 ea.

OTHER CONNECTORS AVAILABLE

.100" Contact Ctrs., .140" Row Spacing.

22/44 Dip Sold.	$2.30 ea.	5 pcs.	$2.10 ea.
25/50 Sold. Eye.	2.95 ea.	5 pcs.	2.75 ea.
40/80 Sold. Eye.	4.80 ea.	5 pcs.	4.50 ea.
43/86 Dip Sold.	4.90 ea.	5 pcs.	4.70 ea.
43/86 Sold. Eye.	4.90 ea.	5 pcs.	4.70 ea.

.156" Contact Ctrs., .140" Row Spacing.

6/– Sgle. Row (PET)	$1.00 ea.	5 pcs.	$0.90 ea.
22/44 Sold. Eye. (KIM)	1.90 ea.	5 pcs.	1.80 ea.
22/44 Dip Sold. (KIM)	1.90 ea.	5 pcs.	1.80 ea.
43/86 Dip Sold.	4.90 ea.	5 pcs.	4.70 ea.

.156" Contact Ctrs., .200" Row Spacing.

15/30 W/Wrap 3	$1.05 ea.	5 pcs.	$0.95 ea.
22/44 W/Wrap 3	2.30 ea.	5 pcs.	2.10 ea.
36/72 Sold. Eye.	3.45 ea.	5 pcs.	3.30 ea.
36/72 W/Wrap 3	3.85 ea.	5 pcs.	3.70 ea.
43/86 W/Wrap 3	5.50 ea.	5 pcs.	5.00 ea.

POLARIZING KEYS FOR ALL OF THE ABOVE:
Specify: IN Contact or BETWEEN Contact:
1 to 49 pcs. $0.10 ea. 50 pcs./Up $0.08 ea.

SPECIAL
12/24 Pin .156" Cont./Ctrs. .200" Row Spacing.
TIN PLATED CONTACTS.
IDEAL FOR PET INTERFACE & PARALLEL USER PORT.
$1.25 ea. 5 pcs. $1.10 ea.

SUBMINIATURE CONNECTORS: (DB 25 SERIES, RS 232.)

DB 25P Male Plug	$2.50 ea.	5 pcs.	$2.20 ea.
DB 25S Female Socket	3.60 ea.	5 pcs.	3.40 ea.
DB 51212-1 Grey Hood	1.20 ea.	5 pcs.	1.10 ea.
DB 51226-1A Black Hood	1.30 ea.	5 pcs.	1.20 ea.
D 20418-2 Hardware Set	0.75 ea.	5 pcs.	0.70 ea.

SAVE: BUY A SET: (1 DB25P, 1 DB25S, Any Hood.)
1 Set: $6.35 ea. 5 sets: $6.15 ea.
NOTE: For Hardware, (D20418-2) Add $.65/Set.

WHISPER FANS
Excellent for computer cabinet cooling. This is the most quiet fan you will find. Only measures 4 3/4" square by 1½" deep. U. L. Listed.
$21.00 ea. 5 pcs. $19.00 ea.

I. C. SOCKETS. GOLD. WIRE WRAP 3 TURN.
14 pin $0.36 ea.
16 pin 0.38 ea.

I. C. SOCKETS. Dip Solder. Tin.
14 pin $0.15 ea.
16 pin 0.17 ea.

2708 EPROMS PRIME
$14.00 ea.

8080 PRIME
$9.00 ea.

WRITE FOR LARGER QUANTITY DISCOUNTS. DEALER INQUIRIES ARE WELCOME.

WE ARE CONNECTOR (EDGE CARD) SPECIALISTS. IF YOU DO NOT SEE WHAT YOU NEED IN THIS ADVERTISEMENT, PLEASE WRITE US. WE WILL REPLY.

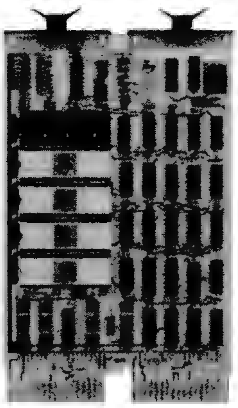
TERMS: Minimum Order $10.00: Add $1.25 for handling and shipping. All orders over $25.00 in USA and Canada: WE PAY THE SHIPPING.
NOTE: CA residents please add 6% sales tax.
NO C.O.D. SHIPMENTS OR ORDERS ACCEPTED.

MAIL ORDERS TO: ***Beckian Enterprises***
P.O. Box 3089
Simi Valley, CA 93063

Discount Prices on

DEC* LSI-11 COMPONENTS

Manufactured and tested by Digital Equipment Corporation. Compatible with DEC PDP*-11/03 and Heathkit H-11. Or build your own computer using the card cage assembly. All cards 8.5 x 5 in.



KD11-HA LSI-11/2 Central Processor Unit. Includes power fail/auto restart, I/O DMA port, real time clock input, vector interrupt handling, firmware debugging and ASCII console routines, List $695 .... $ 599

MSV11-DA 4k x 16-bit RAM, List $416 .... $ 350

MSV11-DB 8k x 16-bit RAM, List $850 .... $ 675

MSV11-DC 16k x 16-bit RAM, List $1375 .... $1095

MSV11-DD 32k x 16-bit RAM, List $2400 .... $1925

H9281-BB 8-slot card cage and backplane .... $ 155

Other components available, write for catalog. Quantity discounts. Terms COD or prepaid.

MICROPROGRAMMING, INC.
1351 Larc Industrial Blvd.
Burnsville, MN 55337
Phone: (612) 894-3510

*Trademark of Digital Equipment Corporation

Unclassified Ads

WANTED: My 4 K PET needs more memory. Anyone with access to eight MOS 6550 programmable memories please write Barry Swartz, 3727 Tartan Ln, Houston TX 77025, (713) 663-6401.

FOR SALE: Altair factory assembled 2SIO board. Wired for TTY and RS232. Never used. Best offer. Write or call Robert Cardamone, 304 S Penn St, Punxsutawney PA 15767, (814) 938-4185.

FOR SALE: RS 232 interface for Diablo printer with CDC interface advertised in May and June 1978 BYTE. $250. H Stone, 64 Morgan Cir, Amherst MA 01002.

FOR SALE: Set of BYTE magazines September 1975, number 1 to July 78, volume 3, number 7 (complete, except November 1977, volume 2, number 11). 33 magazines total. Excellent condition. Best offer, (614) 389-3452.

WANTED: For MBT Inc (model 015) Disk Memory Unit: specifications, schematics and maintenance manual. This company is now out of business: I have been unable to obtain technical information through normal channels. Bert Richardson, 13 Fern St, Natick MA 01760.

FOR SALE: Kleinschmidt teletypewriter TT-117/FG with reperforator/transmitter, Baudot code. Used and working, $100. Tektronix type CA dual trace plug-in unit for 585 or similar scope, $150. M H Research R100B plus and minus 300 V DC power supply, $25. Locarte 3 A 120 V DC power supply (not isolated from AC line), $20. Lambda LT 2095M 0 to 32 V, 0 to 2 A power supply, $75. SwTPC 143 0 to 35 V, 0 to 2 A power supply, $20. All items plus shipping. S Lei, POB 5312, Fargo ND 58102.

FOR SALE: BYTE number 1 to date, $25 year. SCELBI 8B users manual, machine language, assembler, monitor, editor, four issues *SCELBI Computer Digest* for 8008 and similar computers, all for $50. Other manuals and books on electronics and microcomputers. You pay postage. Davey B Moyers, 10743 Karen Gale Ln, Jacksonville FL 32225, (904) 641-9485.

FOR SALE: Texas Instruments TI-59 programmable calculator and PC-100A printer complete with extra paper rolls, magnetic cards and programming manuals. All for $300. Peter Ludwig, 921 Fernwood Av, Plainfield NJ 07062, (201) 263-0200 (ext 3576) days or (201) 753-9780 evenings.

FOR SALE: Heathkit H-8 Computer System; expertly assembled and tested. Includes 24 K static programmable memory, serial and cassette input/output, H-9 video terminal, cassette player, all standard Heathkit software plus Extended BASIC, all documentation. I am graduating from college and must sell; asking $1725 and I will ship it. Call or write Dan Harrington, 927 J St, #49B, Davis CA 95616, (916) 756-7932.

NEW UNCLASSIFIED POLICY

Readers who have equipment, software or other items to buy, sell or swap should send in a clearly typed notice to that effect. To be considered for publication, an advertisement must be clearly noncommercial, typed double spaced on plain white paper, contain 75 words or less, and include complete name and address information.

These notices are free of charge and will be printed one time only on a space available basis. Notices can be accepted from individuals or bona fide computer users clubs only. We can engage in no correspondence on these and your confirmation of placement is appearance in an issue of BYTE.

Please note that it may take three or four months for an ad to appear in the magazine.

FREE: Data cable with the sale of a Persci 1070 intelligent disk controller $500 assembled and tested. TDL SMB board (dealer demonstration model). Two serial input/outputs, one parallel cassette interface and TDL 2 K Zapple Monitor in read only memory and 2 K programmable memory. Fully assembled and tested, $220. TDL software package A with 12 K BASIC, Z-80 assembler, Z-TEL, text output processor, all to run under CP/M format. 8 inch diskette, manuals and notebook, $189. Call or write Ted Nakamura, 3421 Onyx St, Torrance CA 90503, (213) 371-8138.

FOR SALE: Best prices. S-100 bus 16 by 64 video interface (ASCII and block graphics), assembled $100. 8 K programmable memory, assembled $100. Prototyping board, including buffers and regulator on board $20. Full ASCII keyboard, including user defined keys $35. Call or write Philip Klein, 1524 Sacramento St Berkeley CA 94702, (415) 524-9711.

FOR SALE: HP-67 calculator for sale, programmable, automemory stack and much more. Unused, will accept reasonable bid. Call (913) 642-4663.

FOR SALE: MITS Altair 8800A, 8 K programmable memory, serial input/output and audio cassette IO, all documentation, $450. Bright 2610 magnetic tape drive, 800 bits per inch, 27.5 inches per second with two controllers for Data General line of minicomputers. Tape unit was working with one of the controllers when removed from system, condition of other controller unknown. Schematics and manuals included, $850. L D Stricklan, 21733 Alcazar, Monta Vista CA 95014, (408) 257-4805.

FOR SALE: Dual trace oscilloscope, Heath IO-4510, 15 MHz, with calibrator, manuals. Like new, $545. Digital Multimeter, Heath IM-2202 with manual. Like new, $160. Swan 350 Amateur Transceiver, with 117cx power supply, transmitter control unit VFO, manuals. Good condition, $345. Robert Shostak, 1961 Camino de los Robles, Menlo Park CA 94025, (415) 326-0443.

WANTED: Any information that you may have pertaining to Radio Shack TRS-80 software and peripherals. I have a 16 K Level II system and would like to add a printer and disk. Both must be capable of running off a 50 cycle power as I am stationed in Athens Greece. Robert Daniel, PSC Box 2088, APO, New York NY 09223.

FOR SALE: Teletype 3320 printer with 5JE punch/reader, Carterphone DX 103A-7 and Dal-Data Dialer; $730 plus shipping. Steven Terharr, 650 Beech, Moorhead MN 56560, (218) 236-8129.

FOR SALE: Centronics 101, 132 columns, 5 by 7, 165 characters per second, 8 bit parallel interface, uses standard size pin feed paper. $950 on trade for Diablo Hytype II. Frank Bennett, (408) 732-3800 ext 633 (work) or (714) 735-0549 weekends.

OSI CASSETTE INTERFACE USERS: I have some modules for 430 board to increase reliability. Also mods for 420C for standby power and modification of the OSI Audio Cassette Tape Generator program to produce double speed tapes. I would like to share information on these or similar subjects. Phil Bryan, 529 West St, Park City IL 60085.

FOR SALE: TI 59 calculator with all standard accesories and numerous games, in perfect condition, asking $220. Also for sale one slightly used KIM-1, in very good condition with power supply and all manuals. Philip Kaaret, 1113 E State St, Ithaca NY 14850, (607) 272-9119.

FOR SALE: SwTPC 40 printer; 40 columns, 75 lines per minute. I've been using it for about one year and it works great!! $200 or best offer. Digital Group Phi-Deck mass storage with two drives; PHIMON, cabinet and cable. Complete documentation included, $400 or best offer. Holden Caine, 1 Windsor Pl, Melville NY 11746, (516) 692-9512.

FOR SALE: 8 K PET, Commodore. Too small for my needs. Two months old and completely burned in. Some software and additional manuals. Will ship anywhere in USA, $750. Mike Avelis, 108 Wynola Av, New Britain CT 06051, (203) 224-7016 after 6 PM EST.

WANTED: Z-80 or 8080 system monitor and S-100 complete front panel with any available data. Steven Friedel, 33-44 149 St, Flushing NY 11354, (212) 358-8160.

FOR SALE: Sphere Boards: processor/2, cathode ray tube/1A. Best offer. Richard Likwartz, 827 West St, Rock Springs WY 82901, (307) 362-5316.

FOR SALE: 16 K Level II TRS-80, Diablo 1200 Hy-Type (without keyboards), Regency HR-212, R-390 receiver. Will swap 212 for talkie with keypad. Karl Schneider, 4423 W Broadway, Muskogee OK 74401, (918) 683-6511.

FOR SALE: KSR 35 tabletop ASCII teleprinter. Uses either serial or parallel input/output, $550. Four SwTPC 4 K static memory boards, $75 each. Heath IO 101 vectorscope/color bar generator, $85. Gary Wachter, POB 18955, San Antonio TX 78218, (512) 655-9314.

FOR SALE: E & L Instruments MMD-1. Assembled and running, with *Bugbooks*. $250. Jim A Church, 3570 Cortez Dr, Dallas TX 75220.

FOR SALE: 4 K Dynamic programmable memories, MK4096, MK4027, TMS4060, tinned, speed unknown (coded for Honeywell), $2 and $1 each. Teletypes: 35KSR, 33ASR, $450 each. Two Digitronics 3500 HS paper tape readers, less electronics, $75 each. Two MFE Digital cassette transports and parts, $90 for lot. 14 pin DIP cables, $1 each. Will consider trades for Shugart Floppy disks for above. Carl D Cole, 1134 E Geneva Dr, Tempe AZ 85282.

FOR SALE: DEC PDP-8F Minicomputer system. 16 K, two disk drives, video terminal, ASR 33 and much software. Make offer. John Robinson, 725 Berry Ln, Lexington KY 40502, (606) 266-1509.

FOR SALE: Kleinschmit ASCII print drum (76 columns). Electronics/parts for 311-321 printer. Best offer takes it. Bill Vaughn, 2415 Richview Ct, Garland TX 75040, (214) 495-2371 evenings.

WANTED: Software for the VIM-1 on cassette or listings. N Carr, 13709 Peyton Dr, Dallas TX 75240.

BOWLERS NEED HELP: Hardware, software and information needed to operate a state bowling tournament. Robert Woods, 220 Madison St, State College PA 16801, (814) 238-3816.

BYTE ISSUES: I have BYTE numbers 1 thru 15, except number 11. Best offer takes them. Thomas G McBride, 178 Mitchell St, West Orange NJ 07052.

FOR SALE: Heath H8, H9, 16 K, cassette recorder, assembled and running; $1200. David J Marcus, 430 Wolf Hill Rd, Dix Hills NY 11746, (516) 427-1926.

FOR SALE: SwTPC 6800 mainframe; no cards, just mother board and power supply. Assembled and tested. $150. James VanProoyen, Weeks Electric Co Inc, 1057 Cottage Grove SE, Grand Rapids MI 49507, (616) 243-8866.

FOR SALE: AMI EVK-300 system M6800 1 card computer. With 1 K programmable memory, 2 K erasable read only memory and programmer, Proto monitor, Microassembler/Disas read only memory. Four parallel and one serial ports to 19,200 bps, $700. 16 K board for the above, $390. Zvi Peshkess, (517) 355-3164.

FOR SALE: KIM I and power supply; complete, working. First $200 takes it and I will ship it. Send SASE for return of late checks. Send certified check or money order to Judy Upchurch, 107-G Tall Oaks Dr, Greensboro NC 27408.

APPLE OWNERS: I am a collector of Apple software and have over 200 programs for the Apple II. Send me your programs on disk or cassette and I will trade them on a one-to-one basis. Dave Garson, 5163 Willow Wood Rd, Rolling Hills Estates CA 90274, (213) 378-3823.

FOR SALE: Digital Group Z-80 system. 34 K; four digital cassette drives (Phidecks); keyboard; monitor; all software (MaxiBASIC, Business BASIC, assembler, Star Trek, chess, etc). Total price as kit over $3300. Up and running for $3100. John Case, 6703 Timberhill, San Antonio TX 78238, (512) 681-7504.

FOR SALE: SwTPC 6800 Computer System; 16 K programmable memory, serial IO, AC30 cassette interface and Smoke Signal Broadcasting BFD-68 disk system. All documentation and software. $1400 or best offer. Craig Colvin, 817 Cheyenne Dr, Walnut Creek CA 94598, (415) 937-0778.

FOR SALE: What am I offered? BYTE magazines, September 1975 to December 1976 in BYTE binder and full set of 1977 copies, unbound. All perfect condition. Dick Neish, W0SIR, 904 Marday, Sioux Falls SD 57103.

FOR SALE OR TRADE: BYTE volume 1, # 1 through # 10. All ten issues, top condition, $75 or best offer. Don Erickson, 6059 Essex St, Riverside CA 92504, (714) 738-3709 anytime.

WANTED: A Flexowriter with upper and lower case characters suitable for computer generated letters. Working or not. Don Erickson, 6059 Essex St, Riverside CA 92504, (714) 738-3709 anytime.

FOR SALE: MOS technology KIM-1 microcomputer, manuals and power supply included, $150. KIM-3 8 K memory board, manual and power supply included, $100. Martin Goldberger, 15 West 72 St, New York NY 10023, (212) 874-3176 evenings after 6 PM and weekends.

HELP! I was too ambitious. I have 20 M6800 chips in original factory packages. Will sell for $15 each or trade all 20 for a minifloppy drive in good condition. Bill Ganoe, 1634 E Drachman, Tucson AZ 85719.

GTE DIABLO FOR SALE: Commercial Hytype I printer/keyboard, fully equipped; RS232 interface. E Grossman, 410 Albany Post Rd, Croton NY 10520.

FOR SALE: Complete Poly 88 system with a 41 K programmable memory, 3 K erasable programmable memory, 8080A processor, 16 by 64 video, graphics, hardware scroll, 300 and 2400 bps cassette, real time clock, serial port, four parallel ports, single step hardware, 2708 erasable programmable memory programmer, two DACs, 16 channel ADC, 9 inch monitor, cassette deck, all hardware, documentation, and extensive software support. There is one S-100 slot left for a card of your own. I will consider any offer over $2200. Michael Dunn, 45 Livingston Rd # 501, Scarborough, Ontario CANADA M1E 1K8, (416) 266-1635.

INFORMATION WANTED: For G E TDM 114A40 data set: I bought one and I need to test it, repair (if necessary), and use it. I would be willing to trade printing or programming or pay for information, etc. Write Robert Heller, Box 51A Star Route, Wendell MA 01379 (no phone).

FOR SALE OR TRADE: Heathkit Digital Techniques Course and Trainer, Model 15 Teletype parts including keyboards, type boxes, mainframes, and smaller parts. Will trade for Heathkit microprocessor trainer. George Kelm, POB 160, Yap Caroline Is, GUAM 96943.

FOR SALE: Altair 8800 mainframe with processor board, 1 K programmable memory, 2 K programmable memory board. Ideal for hardware oriented beginner. Asking $250. Randy Soderstrom, 4601 Goldfinch, Madison WI 53714, (608) 222-8056.

FOR SALE: Digital Group Z-80 4 board system including processor, IO, video terminal cassette and mother board assembled. Also two 8 K memory boards with ICs and sockets plus MaxiBASIC, Editor, Super/Clock software plus ASCII Keyboard and Encoder. Will take best offer. Asking $500. Oldrich Laznicka, 24 Payson Rd, Belmont MA 02178, or call (617) 484-4978 after 6 PM weekdays or weekends.

WANTED TO BUY: Texas Instrument model SR-52 programmable calculator in good condition. Dale Sebok, 127 Timothy Dr, Tallmadge OH 44278, (216) 633-4297.

FOR SALE: Digital Group complete Z-80 PhiDeck system with speech synthesizer. Major components include dual PhiDecks, monitor, keyboard, 32 K static memory, finished cabinet, VOTRAX speech synthesizer, and substantial software. Assembled and perfect working order. Best offer over $3000 (price new $3750). John Theys, 24 Walnut Av, E Setauket NY 11733, (516) 473-4142.

FOR SWAP: Will swap BYTE issues May, November or December 1977, for September 1976 or January, February or April 1977. Bert Honroe, Schuermanslaan 65, 3070 Kortenberg BELGIUM.

IBM 3705 BSM: Did anyone ever try to connect a 3705-1 Bridge storage module (core). They are available now as most installed model 1s get converted to model 3s having FET storage. If anyone did, please get in touch. I'm having problem in the sense latch. Bert Honroe, Schuermanslaan 65, 3070 Kortenberg BELGIUM.

FOR SALE: Heathkit H9 video terminal, and Heathkit H10 paper reader/punch with parallel board and connector cable. Fully assembled and running, checked out by factory. Best offer for both or either. R Nicosia, 234 41st St, Lindenhurst NY 11757.

FOR SALE: Two Innovex 200 double density 8 inch disk drives, $200 each. Three Innovex 420 double density 8 inch disk drives, $300 each. 4 board Z-80 System (Z-80 4BD) largely factory assembled with 10 A power supply, $900. CAS & CB2 Phi Deck Drive System (kit) with PHI-F, $300. Two blank 8 K memory boards. One IO-4 kit. Robert Frieden, 359 Wilson Av, Kent OH 44240, (216) 673-7181.

FOR SALE: Complete set of BYTE from volume 1, #1 to December 1977. Perfect condition. Jim Larus, 27 Varick Hill Rd, Waban MA 02168.

FOR SALE: Digital Group TVC-64 board; operates fine, with OP-System and documentation. Also Radio Shack keyboard video terminal added. $175 for both. Bob Howarth Jr, RFD # 1, Box 36, Lisbon NH 03585.

SHERLOCK HOLMES FANS: You are invited to correspond with Ben Fairbank, 307 Kent Av, El Paso Texas 79922 to consider together the possibility of undertaking various computer analyses of the "Sherlockian Canon" with the eventual goal of producing a concordance to the 60 stories.

FOR SALE: Heathkit Computer System includes H8 computer, H9 video terminal, 16 K memory, IO board, cassette player, all manuals, schematics. Also includes many programs. Completely assembled and tested. Will pay shipping. $1500. T E Allen, MAG-11, H$MS-11, W/C 620, MCAS Eltoro CA 92709.

FOR SALE: Altair 8800B System: full front panel, 12 K static programmable memory, audio cassette, serial IO, Microterm act-1 keyboard with Sanyo monitor, cassette recorder, package II assembler and 8 K BASIC plus many programs. System fully operational. Total price: $1850. Additional equipment: Altair parallel IO 88-4PIO, $75; Cromemco dazzler, $200; Cromemco bytesaver with 1 K erasable read only memory, $100. Complete documentation. Call Tom, (614) 369-3866 nights.

FOR SALE: 20 back issues of BYTE (from February 1976). Best offer takes all. Norman G Church, 18310 Franklin Way, Gladstone OR 97027, (503) 659-6763.

MUST SELL: SOL 20 with 24 K programmable memory; Panasonic monitor, cassette; North Star controller with Shugart SA400 disk drive; music interface board; 15 K extended cassette BASIC; 12 K disk BASIC; game pack 1. Everything works—less than six months old. $3400 value for $2995. J Andrews, 6303 Kury, Houston TX 77008, (713) 869-3985 evenings.

FOR SALE: 8 K byte, 250 nsec, S-100, static memories, $175. IBM Selectric type balls, $10. IBM Selectric tool kit, $25. IBM Selectric 10 pitch to 12 pitch conversion kit, $35. AC/DC power supply, 5 to 9 V, 20 A, overvoltage protect, overcurrent protect, $80. Sunny power supply, 9 V 25 A, ± 18 V – 4 A; – 9 V – 4 A, $85. Stan Levine, 1802 Melville St, Ocean NJ 07712, (201) 531-8305.

FRIEDEN EQUIPMENT: 36 pieces, one alloter, eight regens, (two brand new—still in crates), two card punches, 13 2305 slave printers, one SPD, SPD stand, one Computyper with program blocks and desk, one Selectadata transmitter, three transmitters, six power supplies, miscellaneous assortment of cables, spare parts, schematics, paper tape and edge punch cards. Also three McGraw Edison Voicewriters. No reasonable offer refused, plus shipping. Ron Komara, POB 267, Davidsville PA 15928, (814) 479-4674.

FOR SALE: SOL 20, two SOL 10s, Altair 8800A with mother board, 24 K bytes of static programmable memory, IO board, MITS vector interrupt board, real time clock board, ICOM dual floppy disks, ASR 33 Teletype, TDL Z-processor. Send offers and receive detailed list. Herbie Marsden, 608 Kelly, Silver City NM 88061, (505) 538-5229.

FOR SALE: ASR 33 like new, with recent Integrated Circuit Touch Tone Modem. Includes all manuals, $750. You ship. Also SwTPC CT-1024 with scroll, 16 lines, 64 characters, upper/lower case, custom oak/formica cabinet, and lots of spare parts. $225. Julian E Jetzer, 6400 Hawthorn Rd, Sheboygan WI 53081, (414) 457-3366.

FOR SALE: Poly 88 chassis with two Altair 4 K static memory boards, WAMECO 8080 processor board, Vector Graphic Reset-n-go programmable and read only memory board and National Multiplex IO board with 4800 bps digital cassette deck. All are in excellent working condition. Will include nonworking S D Sales Z-80 processor board. Sell system for $500 or will sell separately. W R Giffen, POB 781, Richardson TX 75080.

FOR SALE: PDP 8L minicomputer with 4 K core and teletype interfaces. I Ehrlich, 284 Hendrix St, Philadelphia PA 19116.

FOR SALE: Wintek M6800 processor with ASCII keyboard, ACIA, monitor, two PIAs, 5 K memory/power supplies, and cassette tape IO. $250 or best offer. Bob Watson, 2853 Pebble Beech Dr, Flagstaff AZ 86001, (602) 526-2312.

FOR SALE: AKI keyboard, matrix encoded, power supply and 5 level paper tape punch, $50. 5 level paper tape reader, $50. Ron Rogers, POB 17147, Baton Rouge LA 70893.

FOR SALE: MMD-1 8080-based system for interfacing experimentation and software development. Assembled and tested, $275. Norm Levin, 4408 Sherwood Rd, Philadelphia PA 19131.

WANTED: Cylindrical slide rule (such as Thatcher or Fuller) and pocket circular slide rule (such as Carpenter or Sperry) or any other unusual old slide rule. Also need pocket mechanical calculator (Curta). Describe and price. Dr George Wentz, POB 626, San Marcos TX 78666, (512) 392-2872 after 7 PM.

FOR SALE: Centronics Printer # 100; used, $750. Aaron Epstein, 5437 Laurel Canyon Blvd, Suite 208, N Hollywood CA 91607, (213) 762-0020.

FOR SALE: Ithaca Audio Z-80 board, $35; Percom Data CI-812 cassette interface, $30; D C Hayes board, $50. All bare boards with sockets installed. TDL Macroassembler, text output program, Z-Tel, Zapple, text editor, $120 for all software. Kim, Calgary CANADA, 283-6863.

Reader Service

To get further information on the products advertised in BYTE, fill out the reader service card with your name and address. Then circle the appropriate numbers for the advertisers you select from the list. Add a 15 cent stamp to the card, then drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by BYTE. This helps us bring you a bigger BYTE.

*Correspond directly with company.

BOMB – BYTE's Ongoing Monitor Box

Pascal Blazes Into First Place

"A 'Tiny' Pascal Compiler, Part 1," page 58, by Chung and Yuen placed first in the September BOMB. Second place went to "WADUZITDO," page 166, by Larry Kheriaty. These articles placed 2.0 and 1.5 standard deviations above the mean, respectively; first and second prizes of $100 and $50 will be sent to the authors. In third place was "The Mathematics of Computer Graphics," page 22, followed by "Graphic Manipulations Using Matrices," page 156, in fourth.

The BOMB (BYTE's Ongoing Monitor Box) is our monthly forum in which readers rate the articles in BYTE. To participate, see the card at right.■

"My 8 to 5 minifloppy™ now works nights and weekends."

"I own a fast-growing business and before I bought my computer system I put in a lot of late hours keeping up with my accounting and inventory control. Now the computer does my number crunching quickly, so I have time after hours to have some fun with the system. My son and I started out playing Star Trek on the system, and now we're learning to play chess.

"When I was shopping around for my system, the guys in the computer stores demonstrated all the unique features of the minifloppy. I've got to admit that at first I didn't really understand all the technical details. But now that I use the system every day, I really appreciate the minifloppy's fast random access and data transfer. I like the reliability, too.

"I'm glad I went with Shugart drives. Look, when you lay out your own money for a system, you want dependable performance and good value. Do what I did. Ask for the system with the minifloppy."

If it isn't Shugart, it isn't minifloppy.

Shugart Associates

435 Oakmead Parkway, Sunnyvale, California 94086

For a list of manufacturers featuring Shugart's minifloppy in their systems, circle reader response number.

TM minifloppy is a registered trademark of Shugart Associates

Circle 312 on inquiry card.